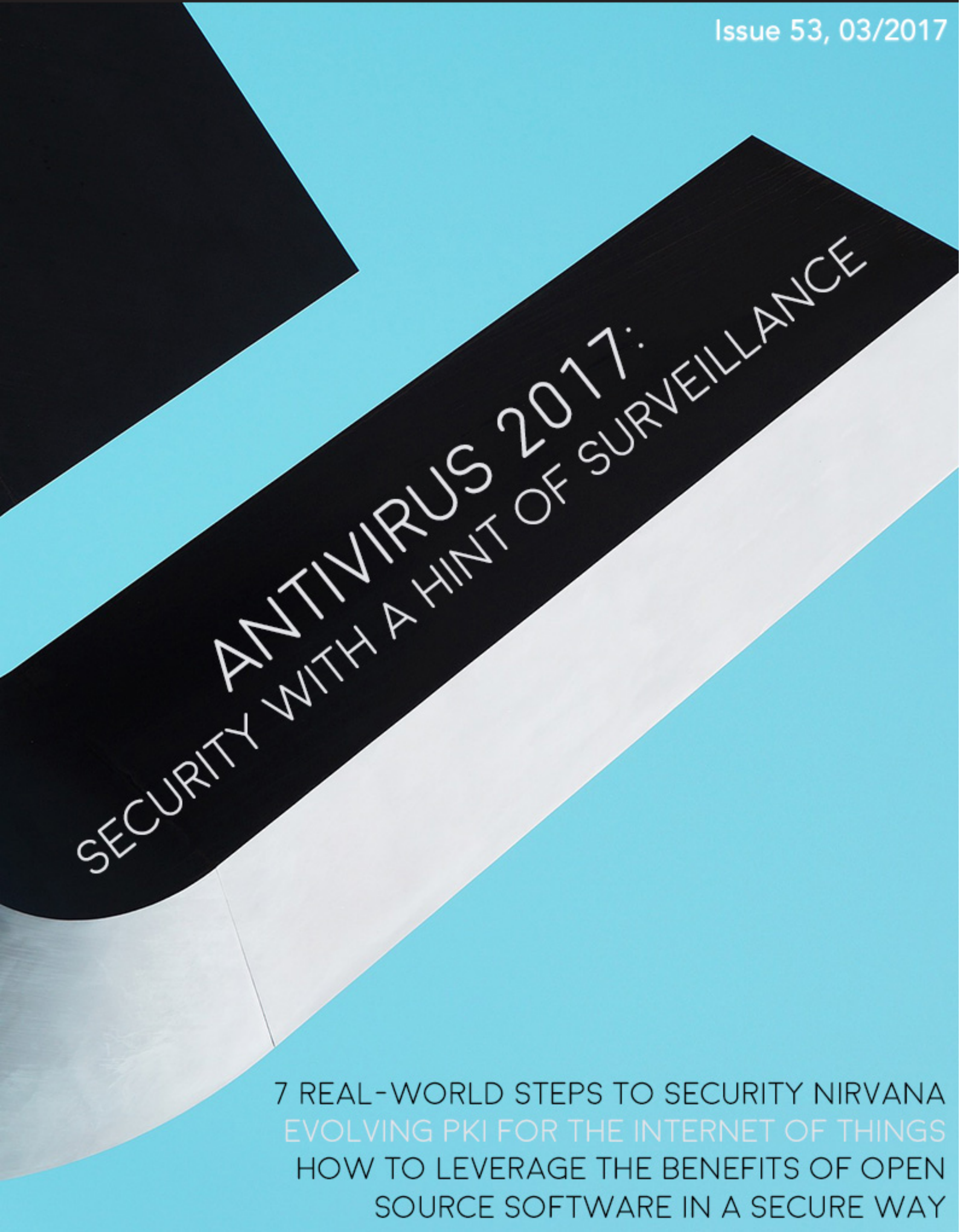
[+] (IN)SECUREMagazine

Issue 53, 03/2017



ANTIVIRUS 2017:
SECURITY WITH A HINT OF SURVEILLANCE

7 REAL-WORLD STEPS TO SECURITY NIRVANA

EVOLVING PKI FOR THE INTERNET OF THINGS

HOW TO LEVERAGE THE BENEFITS OF OPEN SOURCE SOFTWARE IN A SECURE WAY

TABLE OF CONTENTS

(IN)SECURE Magazine 53 CONTRIBUTORS LIST

- **Marnix Dekker**, IT Security Strategy and Policy at the European Commission
- **Colin Domoney**, Senior Product Innovation Manager at Veracode
- **Ken Ivanov**, PhD, Director & Chief Security Expert at EldoS Corporation
- **Bob Janssen**, CTO, founder, and SVP of Innovation of RES
- **Kayne McGladrey**, Director of Information Security Services at Integral Partners
- **Jeremy Rowley**, VP of Emerging Markets at DigiCert.

Visit the magazine website at www.insecuremag.com

Feedback and contributions: Mirko Zorz, Editor in Chief - mzorz@helpnetsecurity.com

News: Zeljka Zorz, Managing Editor - zzorz@helpnetsecurity.com

Marketing: Berislav Kucan, Director of Operations - bkucan@helpnetsecurity.com

Worldwide infosec spending to reach $90 billion in 2017

Enterprises are transforming their security spending strategy in 2017, moving away from prevention-only approaches to focus more on detection and response, according to Gartner.

Worldwide spending on information security is expected to reach $90 billion in 2017, an increase of 7.6 percent over 2016, and to top $113 billion by 2020. Spending on enhancing detection and response capabilities is expected to be a key priority for security buyers through 2020.

“The shift to detection and response approaches spans people, process and technology elements and will drive a majority of security market growth over the next five years,” said Sid Deshpande, principal research analyst at Gartner. “While this does not mean that prevention is unimportant or that CISOs are giving up on preventing security incidents, it sends a clear message that prevention is futile unless it is tied into a detection and response capability.”

Mr. Deshpande said that skills shortages are further driving spending on security services. Many organizations lack established organizational knowledge of detection and response strategies in security because preventive approaches were the most common tactics for decades. Skill sets are scarce and, therefore, remain at a premium, leading organizations to seek external help from security consultants, managed security service providers (MSSPs) and outsourcers.

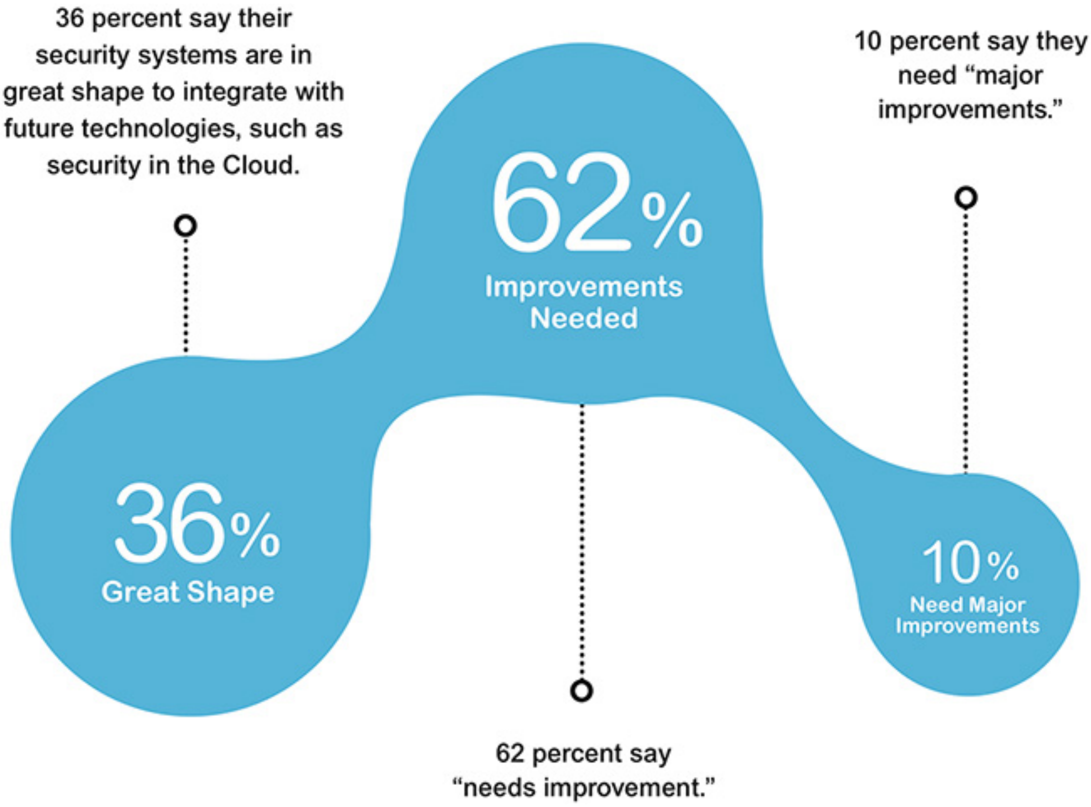
The need to better detect and respond to security incidents has also created new security product segments, such as deception, endpoint detection and response (EDR), software-defined segmentation, cloud access security brokers (CASBs), and user and entity behavior analytics (UEBA). These new segments are creating net new spending, but are also taking spend away from existing segments such as data security, enterprise protection platform (EPP) network security and security information and event management (SIEM).

21% of websites still use insecure SHA-1 certificates

New research from Venafi Labs shows that 21 percent of the world's websites are still using certificates signed with the vulnerable Secure Hash Algorithm, SHA-1.

On February 23, 2017, Google affiliated security researchers announced they cracked the SHA-1 security standard using a collision attack. The incident proved that the deprecated cryptographic secure hash algorithm still used to sign many website digital certificates can be manipulated.

Newly issued certificates using the SHA-2 family of hash functions solve these problems, but Venafi Labs' research shows that many companies have not replaced all their certificates with ones signed by SHA-2. This leaves organizations open to security breaches, compliance problems, and outages that can affect security, availability, reliability and even profits.



Will most security operations transition to the cloud?

Companies across industries are increasingly leveraging the cloud for security applications, with 42 percent indicating they currently run security applications in the cloud and 45 percent stating they are likely or extremely likely to transition security operations to the cloud in the future, according to Schneider Electric.

Organizations utilize the cloud for existing applications including data storage, human resources, email and security, and are eager to continue adopting it for security operations. Fifty seven percent of respondents believe the cloud is secure, with IT and technology professionals having the most confidence (78 percent), followed by education (70 percent), construction (68 percent) and financial services (52 percent). However some skeptics remain, with 18 percent of respondents indicating they do not trust the cloud.

Nearly three-fourths of respondents said network security is an important feature for security systems in their organizations. While the state of security continues to advance, respondents indicate security systems aren't where they should be in order to adopt emerging technologies (54 percent), and despite business leaders being supportive of emerging technology (95 percent), many barriers to adoption exist.

Online banking customers remain extremely frustrated with passwords

A new survey by iovation and Aite Group, polled nearly 1,100 consumers across four generations who use online and/or mobile banking platforms to better understand their attitudes toward various authentication mechanisms used today.

Despite being comfortable using passwords, the study found that 85% of survey respondents recognized the need to bolster online security by moving beyond this increasingly archaic method for authentication. However, due to varying comfort levels and willingness to learn new techniques, different generations expressed varying preferences around the best alternative to replace the ubiquitous password.

A clear correlation emerged between consumers' openness to change and their age as iovation surveyed millennials (35%), Gen X'ers (26%), Baby Boomers (32%) and seniors (7%). The report revealed 95 percent of millennials are open to using something other than a password, as are the majority of Gen X and Baby Boomer respondents (both 82%). And while only 16 percent of seniors are very willing to learn new authentication methods, 48 percent report that they are willing to try a different way.

"There's no denying that passwords can no longer protect online assets the way they are meant to, the way they used to," said Julie Conroy, Research Director for Aite Group's Retail Banking & Payments practice and author of the report.

58% consumers favor passwords because it is what they are most familiar with.

85% of consumers across generations are eager to replace passwords with modern authentication methods.

Kentik delivers 600% growth

Kentik announced continued rapid expansion in 2016, growing 6x in annual recurring revenue and now observing hundreds of Terabits of Internet traffic.

Kentik's breakout year in 2016 included an infusion of new funding, accolades from the major enterprise industry analysts, and a rapidly expanding base of global customers, including GTT, Pandora, Neustar, Yelp, and Box, as well as global Tier 1 telecom service providers, financial services firms, and cloud providers. Kentik released new functionality in its Kentik Detect offering, including accurate network anomaly detection, a cloud-friendly network performance monitoring solution, and an end-to-end DDoS defense solution including integration with Radware and A10 Networks.

"We are extremely proud of our 2016 results, which reflect the market's intense and growing demand for cloud-based network performance management and DDoS defense solutions," said Avi Freedman, CEO of Kentik.

To fuel its continued growth, Kentik received a $23 million Series B funding round in August led by Third Point Ventures, with participation by existing investors August Capital, Data Collective (DCVC), First Round Capital, and Engineering Capital, along with new investors Glynn Capital and David Ulevitch.

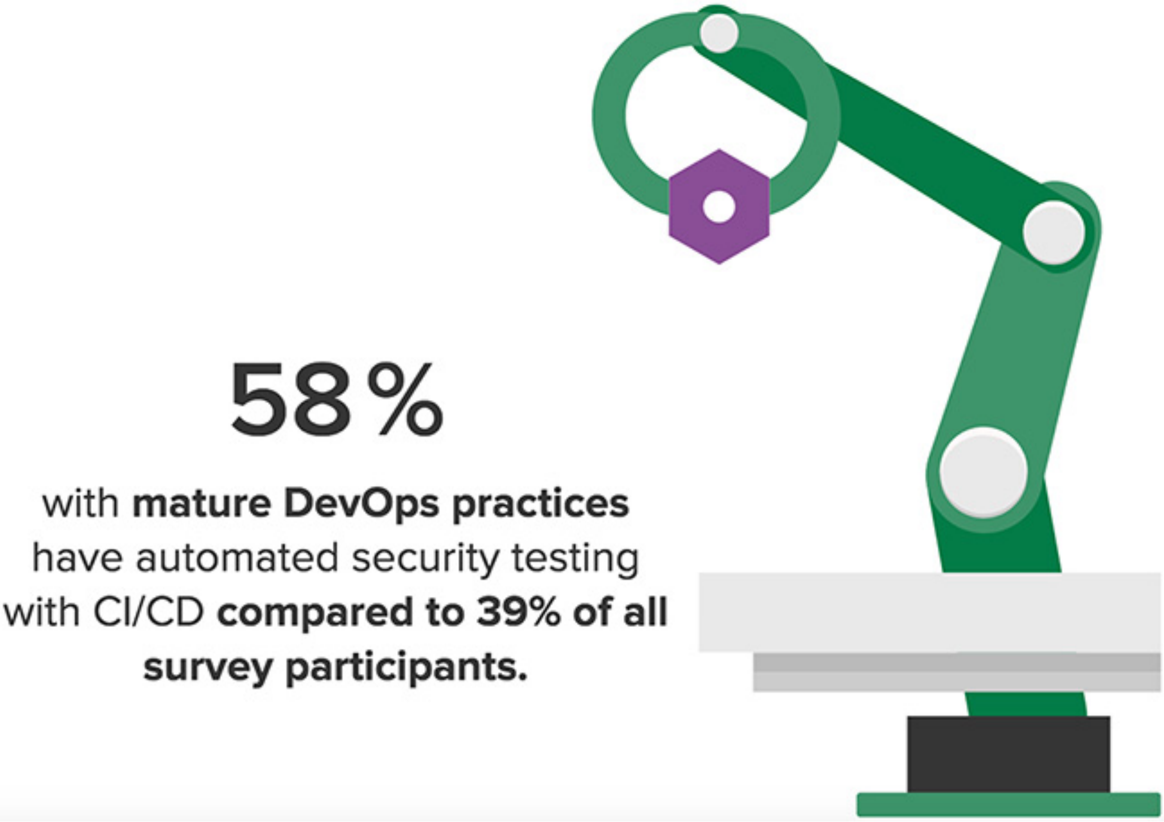
Software development teams embrace DevSecOps automation

Mature development organizations ensure automated security is woven into their DevOps practice, early, everywhere, and at scale, according to Sonatype.

The adoption of DevOps around the world is evidenced by 67% of survey respondents describing their practices as very mature or of improving maturity.

Where traditional development and operations teams see security teams and policies slowing them down (47%), DevOps teams have discovered new ways to integrate security at the speed of development. Only 28% of mature DevOps teams believe they are being slowed by security requirements.

42% of mature DevOps organizations perform application security analysis at every stage of the software delivery lifecycle (SDLC). This number shrinks to just 27% when all survey respondents are counted.


58%
with **mature DevOps practices** have automated security testing with CI/CD **compared to 39% of all survey participants.**

Organizations still vulnerable to brute force attacks

While increases in malware are clearly a major threat to both enterprises and service providers, network complexity is creating its own vulnerability, according to Ixia.

The average enterprise is using six different cloud services, and network segmentation is increasing, yet 54% of enterprises are monitoring less than half of those network segments, and less than 19% of companies believe that their IT teams are adequately trained on the wide array of network appliances they are managing.

Gaining access to accounts is often done the old-fashioned way—brute force guesses, starting with the most obvious. It is shocking how many network accounts and devices contain default usernames and passwords. At the top of the list were usernames like “root” and “admin,” but also “ubnt,” which is the default username for AWS and other cloud service offerings that use Ubuntu. IoT was also a notable target with “pi” for Raspberry PI.

- The top 5 username guesses were: root, admin, ubnt, support, and user.
- The top 5 password guesses were: null, ubnt, admin, 123456, and support.

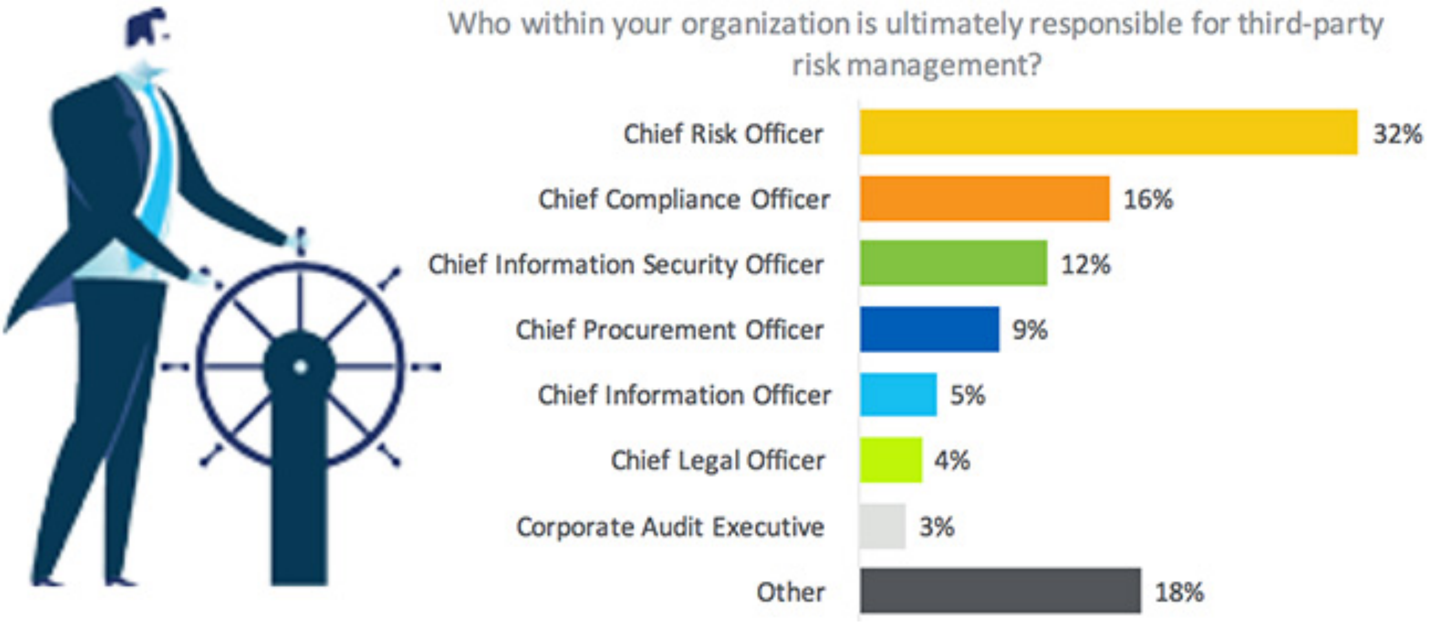
300+ Cisco switches affected by critical bug found in Vault 7 data dump

While combing through WikiLeaks' Vault 7 data dump, Cisco has unearthed a critical vulnerability affecting 300+ of its switches and one gateway that could be exploited to take over the devices.

The flaw is present in the Cisco Cluster Management Protocol (CMP) processing code in Cisco IOS and Cisco IOS XE Software.

"The vulnerability is due to the combination of two factors: the failure to restrict the use of CMP-specific Telnet options only to internal, local communications between cluster members and instead accept and process such options over any Telnet connection to an affected device, and the incorrect processing of malformed CMP-specific Telnet options," Cisco explained.

"An attacker could exploit this vulnerability by sending malformed CMP-specific Telnet options while establishing a Telnet session with an affected Cisco device configured to accept Telnet connections. An exploit could allow an attacker to execute arbitrary code and obtain full control of the device or cause a reload of the affected device."



Managing third-party risk: Dominant trends

One in five organizations has faced significant risk exposure due to a third party in the last 18 months. Of those who shared loss data, 25% said that the loss impact was greater than $10 million.

As companies outsource their processes or services, they expose themselves to a range of third-party risks, including data security risks, business disruptions, legal liabilities, corruption and bribery risks, and compliance risks – all of which have a major impact on profits and brand value.

Fourth-party risk management is also emerging as a key area of focus, with organizations being held responsible not just for the actions of their immediate third parties, but also for the actions of their third parties' vendors and suppliers. Adding further impetus are regulations from authorities such as the Office of the Comptroller of the Currency (OCC) and the Consumer Financial Protection Bureau (CFPB), as well as mandates such as the UK Bribery Act and the Health Insurance Portability and Accountability Act (HIPAA), which stipulate stringent requirements for third-party governance.

21% of respondents reported that their organizations faced risk exposure due to third parties in the last 18 months; of those who shared financial impact data on the losses, 25% said that the loss impact was greater than $10 million.

Intel is offering up to $30,000 for bugs in its hardware

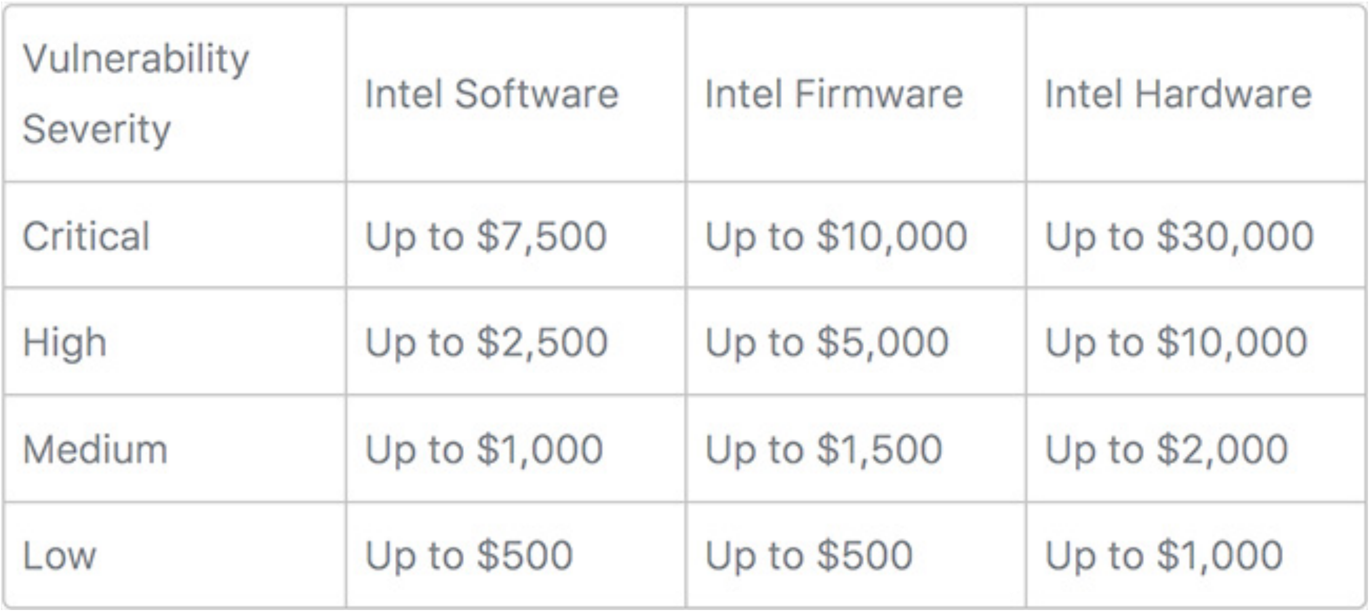
Intel is looking for bug hunters to deliver information about its software, firmware and hardware, and discovered vulnerabilities in the latter will bring the biggest rewards.

Depending on the vulnerability's severity, researchers can earn themselves as much as $30,000.

Of course, Intel will be the judge of how severe is each vulnerability.

The company does not want bug hunters to poke into its Web Infrastructure, third-party products and open source programs.

"Recent acquisitions are not in-scope for the bug bounty program for a minimum period of 6 months after the acquisition is complete," they also noted.



Vulnerability Severity	Intel Software	Intel Firmware	Intel Hardware
Critical	Up to $7,500	Up to $10,000	Up to $30,000
High	Up to $2,500	Up to $5,000	Up to $10,000
Medium	Up to $1,000	Up to $1,500	Up to $2,000
Low	Up to $500	Up to $500	Up to $1,000

Lip movement: Authentication through biometrics you can change

Choosing a unique, complex and long enough password that will still be easy to remember is a big challenge for most users, and most of them would happily opt for biometric authentication in a heartbeat.

But the problem with physical biometrics – fingerprints, palm prints, iris shape, etc. – is that you can't change them if they get compromised. A good solution to that problem might be in the combination of physical and behavioral biometrics and a password.

An elegant and relatively easy to use option is the "lip motion password" – a technology invented by Hong Kong Baptist University computer science professor Cheung Yiu-ming, and patented in the US in 2015.

The technology uses a person's lip motions to create a password, and the system verifies a person's identity by simultaneously checking whether the spoken password and the behavioural characteristics of lip movement match.

The system takes into consideration the lip shape and texture as the user voices (or simply silently mouths) the password, and is able to detect and reject a wrong password uttered by the user or the correct password spoken by an imposter.

"The same password spoken by two persons is different and a learning system can distinguish them," the professor noted. So, even if an attacker knows the password, it's impossible for him or her to use it to successfully impersonate the target. And if, by any chance, the attacker has managed to record a video of a user's lip while he or she was pronouncing the password, a simple change of the actual content of the password is enough to prevent future impersonation.

Java and Flash top list of most outdated programs on users' PCs

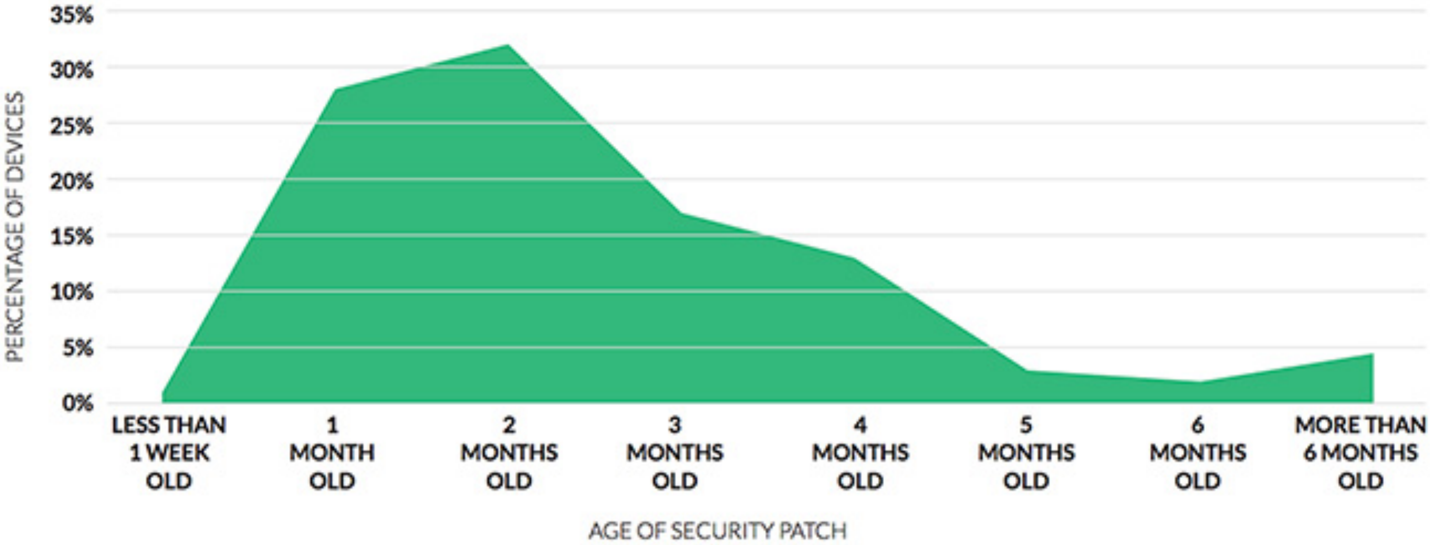
52% of the most popular PC applications, including Flash and Java, are out-of-date. Gathered anonymously from 116 million Windows desktop and laptop users, Avast found the most outdated programs include:

- Java (Runtime 6,7)
- Flash Player (Active X)
- Foxit Reader
- GOM Media Player
- Nitro Pro
- WinZip
- DivX
- Adobe Shockwave Player
- 7-ZIP
- Firefox.

Topping this list of the least updated applications is Java, with more than 24 million people running the outdated versions Java Runtime 6 and 7. And while another 26 million users are on the latest version Java 8, more than 70% haven't installed the latest update rollout (currently update 121).

This is closely followed by Flash, where 99% of users have yet to update this control for Internet Explorer; and Foxit Reader which sees 92% of users working with an old version of the application.

Conversely, the most up-to-date applications are Google Chrome at 88%; Opera at 84%; and Skype which is 76% up-to-date across the sampled user base, which illustrates that even the programs that auto-update are not necessarily always up-to-date.



Lack of security patching leaves mobile users exposed

An analysis of the patch updates among the five leading wireless carriers in the United States found that 71 percent of mobile devices still run on security patches more than two months old. Six percent of devices run patches that are six or more months old. Without the most updated patches, these devices are susceptible to myriad of attacks, including rapidly rising network attacks and new malware, also detailed in the report.

In tech city centers, Boston topped a list of tech cities with the largest growth in network incidents with a more than 960 percent increase. Skycure also found that common malware grew by more than 500% from the first quarter to the fourth quarter of 2016.

A huge number of Android vulnerabilities were identified in 2016, rising to more than four times the number in 2015. Almost half of those vulnerabilities allowed excessive privileges, while others allowed other bad effects, like leakage of information, corrupted memory, or arbitrary code execution. Because carriers must make Android patches available to their users before they can patch their devices, Skycure analyzed devices on AT&T, MetroPCS, Sprint, T-Mobile, and Verizon to determine the age distribution of security patches on the leading carriers.

iStorage launches datAshur Personal²

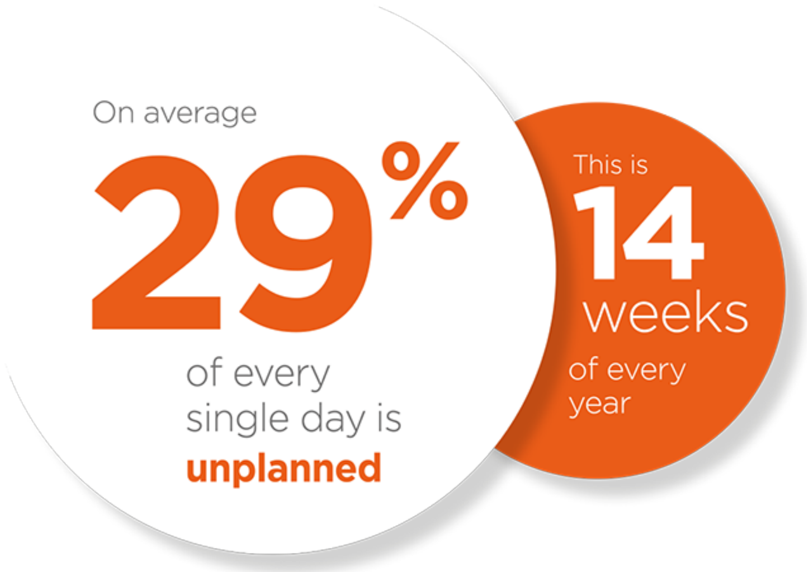
iStorage have launched a new USB 3.0 flash drive for individuals concerned about keeping their personal information safe. The datAshur Personal² has been designed with innovative technology for rapid transfer speeds as well as advanced levels of encryption. It has been designed to be 10 times faster than its predecessor with data transfer rates of up to 116MB/s read and 43 MB/s write. The device is available in 8GB, 16GB, 32Gb & 64GB capacities, and boasts widespread compatibility with all operating systems to include Windows, Mac, Linux and Android systems.

The datAshur Personal² includes an on-board keypad for authentication. The drive can only be unlocked by entering a personal 7-15 digit PIN code before connecting to a USB port. All data transferred to the device is automatically hardware encrypted to military standards, to ensure all data is safe and secure in the event the drive is lost, stolen or tampered with. As soon as the device is unplugged from the computer or when power to the USB port is turned off, the device auto locks keeping the information secure.

IT pros spend too much time handling emergencies

A 1E survey of 1,014 IT professionals, who together manage more than 21 million endpoints globally, centered on unplanned activities – how often they occur, what types are most common, and the time spent identifying and addressing issues.

On average, IT workers spend 29 percent of every day reacting to unplanned incidents or emergencies. Based on a full-time work schedule of 1,700 hours per year, this equates to more than 14 weeks a year.



How to leverage the benefits of open source software in a secure way

By Colin Domoney

In 2011, Marc Andreesen famously said that “software is eating the world.” By 2015, open source software had all but eaten the software world with the release of mainstream solutions such as Apple’s Swift programming language and Microsoft’s .Net framework as open source projects. The 2016 Black Duck Software Future of Open Source survey revealed that 78% of the respondents are now running their businesses with open source software, and two-thirds are building their own products based on open source software.

There are several reasons for the explosion in adoption of open source software. Firstly, open source adoption drives a competitive advantage allowing an organization to deliver innovation and features more quickly by leveraging existing software libraries and frameworks. Secondly, open source adoption attracts top talent, as open source is often seen as superior to proprietary or closed source solutions.

Developer-led innovation vs security

Increasingly, software developers are the ones who determine which software components will be used in the enterprise. Common open source binary repositories are afforded first-class support within developer IDEs (Nuget in Visual Studio, MavenCentral in Eclipse and IntelliJ), allowing developers to ingest open source components into their projects with minimal effort. GitHub is the primary source control platform for open source projects. It has undergone an exponential growth since 2010, and is now hosting over 10 million projects.

Gone are the days when an enterprise would conduct a lengthy due diligence process and vendor review of a proprietary software solution such as an ERP or CRM solution; an enterprise can now unknowingly incorporate open source software of unknown provenance into its core product.

This has made the job of the legal and security professionals much more challenging. If it is no longer possible to review software as it enters the enterprise, how can the necessary governance be put in to place to ensure that said software is free of vulnerabilities and is licensed appropriately?

Is open source software more or less secure than closed source software?

Linus's Law suggests that "given enough eyeballs, all bugs are shallow." In other words, copious peer review will mean all but the least trivial flaws will be discovered in open source software and eliminated in a collaborative effort, akin to the way the Linux kernel was built collaboratively. However, the Heartbleed vulnerability seems to disprove this postulate, since it was not spotted by the legions of open source contributors who scrutinized it.

Jeff Atwood suggests that there are several fallacies in regard to Linus's Law: there is a difference between usage and development eyeballs (in terms of skillset), it is easier to review your own code than someone else's, and there are not enough qualified eyeballs.

Commercial bug bounty programmes are an excellent way to ensure that reviewers are attracted to reviewing critical pieces of open source software. And if open source software is going to be open to public scrutiny and review, doesn't this mean that it will be more easily exploited? And when patches are released, these will be visible to attackers, meaning they will have detailed knowledge of the exact nature of the vulnerability and, hence, can craft a suitable exploit?

A closed source system is inherently no more secure than an open source one. However, with a closed source system the end user is at the mercy of the supplier for a fix or patch. In the case of open source software, suitably motivated users could implement a fix themselves (and contribute it to the community). According to Russell Clarke, David Dorwin, and Rob Nash, ("Is Open Source Software More Secure?", Homeland Security/Cyber Security, 2009), a closed source system can hardly rely on "security through obscurity" as prevention against exploits.

Best practices for the adoption of open source software

In order to ensure a competitive advantage, a modern enterprise is compelled to leverage open source software. However, this should not be at the expense of the security of the product. The following best practices maximize the benefit of open source software:

Patch management

A centralized patch management framework is vital for ensuring that the most critical vendor patches are applied to the infrastructure in a timely fashion. By patching just six software packages it is possible to significantly reduce the likelihood of malware infection.

Many enterprises run a variety of legacy Java runtime environments, and many of the older versions of JRE have severe vulnerabilities - a significant risk reduction is possible by migrating to the most recent versions of JRE.

Prescribe a policy

Organizations will have varying degrees of risk appetite based on their market and maturity. Perhaps time-to-market and innovation is more important than a reduced likelihood of a data breach? It is important that an organization prescribes a policy (or at least a guideline) regarding the utilization of open source software. Otherwise, the development team will assume they are free to use open source of any provenance, and this may result in a product shipped with known vulnerabilities and/or incompatible software licenses.

Once software has been released, it may prove costly and time consuming to retrospectively address any issues surrounding the use of open source components. A sensible and pragmatic policy should forbid the use of software components with a known high rate of vulnerabilities.

Control your repositories

Modern IDEs are optimized around giving developers access to the widest possible selection of open source libraries directly within their native environments. If such a development practice is in contravention of your policies, then it may be necessary to bar access to such repositories either by blocking access at the firewall level or, more pragmatically, by providing an on-premise cached version of known and approved software components.

Various commercial products are designed to provide a local cached version of popular repositories, allowing the security team to closely control which components (and hence which vulnerabilities) are being included into the final product. Additionally, the judicious usage of local repositories ensures that only a single, given approved version of a component is used, rather than a myriad of different (and potentially vulnerable) versions.

Understand your software supply chain

Many organizations are likely to include software from other vendors and COTS components. In such cases, it is possible that both known and unknown vulnerabilities will be inherited into your software supply chain.

The use of security testing tools (both static code analysis and software composition analysis tools) will provide a high degree of visibility into inherent risk, and vendor contracts should be made to mandate a minimum security level for delivered software components into the receiving organization.

Understand how to remediate

An enterprise should continuously assess its risk from vulnerabilities within its open source and third party. When a new risk is detected (e.g. a new vulnerability is disclosed) the security team should proactively work with the development organization to remediate.

Best practices for remediation vary depending on component and product complexity; a simple “upgrade to the latest component” may introduce unintended regression if the component has changed significantly. A pragmatic approach would be to determine whether the vulnerability is actually exploitable, and if so, whether a closely matching non-vulnerable library is available to minimize collateral impact.

Colin Domoney is the Senior Product Innovation Manager at Veracode (www.veracode.com).

Want to reach a large audience of
security pros by writing for (IN)SECURE?
Send your idea to
mzorz@helpnetsecurity.com

Antivirus 2017: Security with a hint of surveillance

By Ken Ivanov

Recently, my dad asked me to look into a connectivity issue on his PC. The issue itself was not a big deal, but what I accidentally discovered while working on it was.

My dad had Kaspersky Internet Security (KIS) - a widely used anti-virus and anti-malware tool - operating on his laptop. The tool, as per its vendor's web site, is supposed to "safeguard you against today's Internet threats, defend your privacy and personal information, and boost security for online shopping and banking." That very well may be, but some activities the tool performed on my dad's computer do not seem entirely above board.

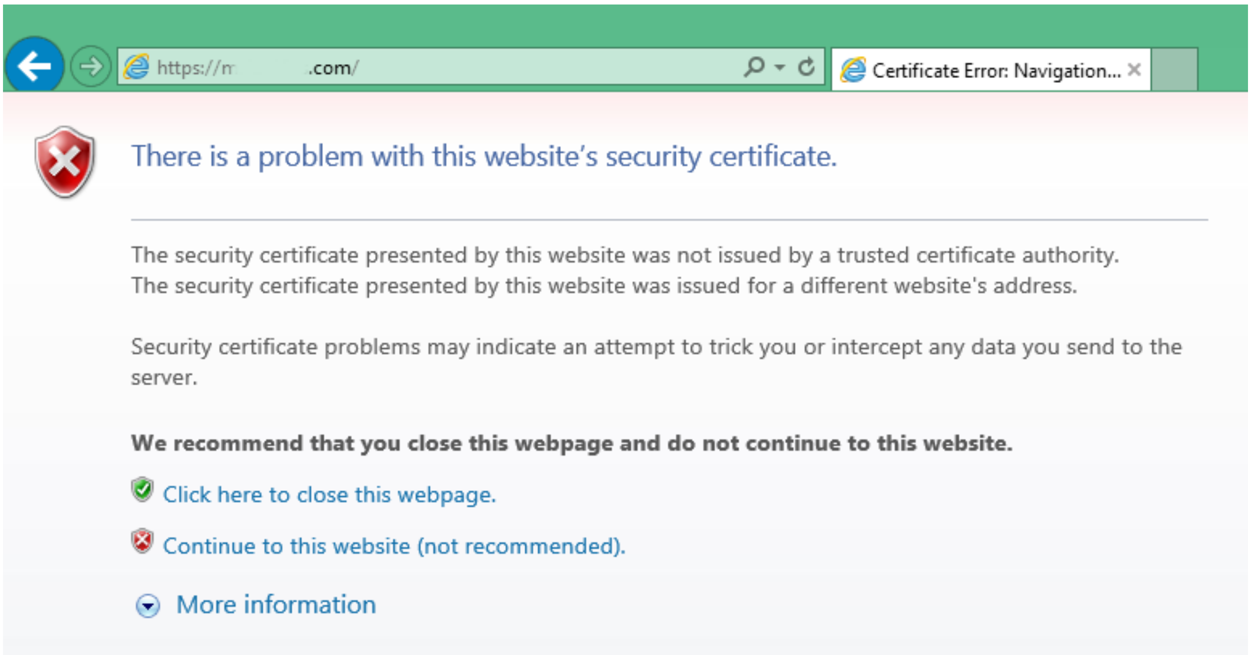
I noticed is that KIS quietly watched over my dad's secure HTTPS traffic. Whenever he connected to an HTTPS web site with his browser, the tool would intercept his request, inject itself in the middle of the connection, and then quietly sit there and relay all the data exchanged between the browser and the web server. Obviously, the tool had access to all that data (encrypted data!), and was free to alter it in any way it liked.

You might note that similar web monitoring techniques have been used by firewalls to look for web-distributed malware for ages, and you would be correct. But, there are two principal differences between the two cases. Firstly, general-purpose malware protection tools have never monitored protected traffic.

HTTPS was specifically designed to provide privacy, authenticity, and protection from surveillance, and as such users employing HTTPS expect their traffic to come from a trusted source and to be protected from prying eyes, whomever those eyes may belong to. Secondly, the technique that KIS used to squeeze itself into the secure channel appeared to me fundamentally wrong and unethical and, what's worse, quite insecure, as it created a number of derivative risks to the safety and privacy of the information belonging to the user.

To understand the depth of the problem we first need to recall the role played by digital certificates in Transport Layer Security (TLS).

Every secure website has a digital certificate associated with it. Every time a browser connects to an HTTPS endpoint, the website sends back its certificate, which the browser inspects in order to establish the website's trustworthiness. If the certificate is up-to-date, matches the website's address, comes from a trusted authority and is not revoked, the browser considers the website to be genuine and goes ahead with the connection. If there's a problem with the certificate, the browser will warn the user about it, and let the user decide what to do next (terminate the connection or go ahead despite the problem).



There is a problem with this website's security certificate.

The security certificate presented by this website was not issued by a trusted certificate authority.
The security certificate presented by this website was issued for a different website's address.

Security certificate problems may indicate an attempt to trick you or intercept any data you send to the server.

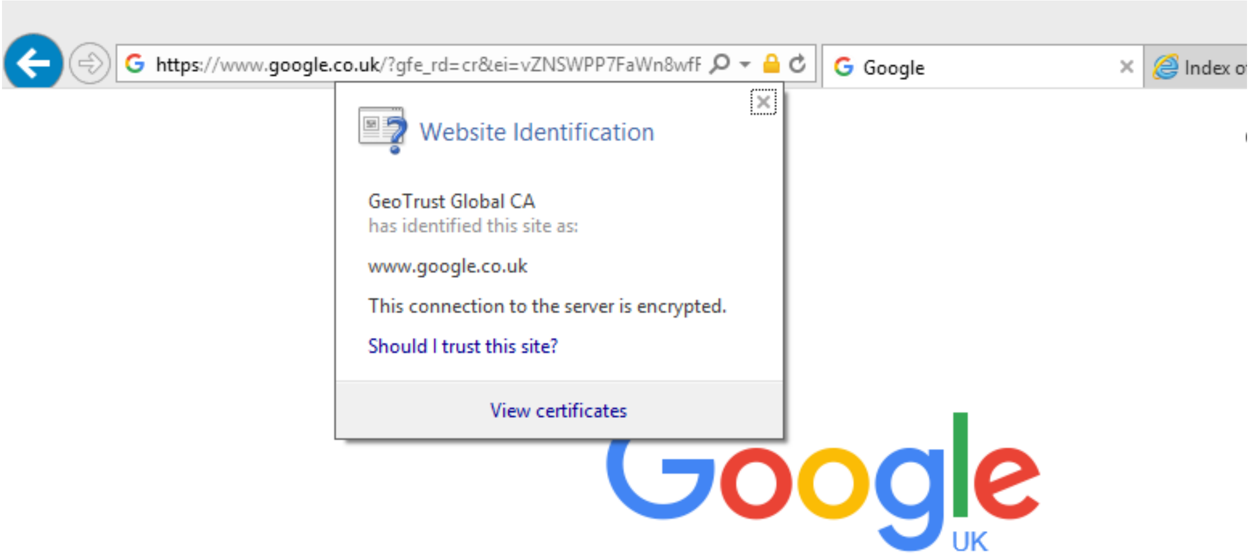
We recommend that you close this webpage and do not continue to this website.

Click here to close this webpage.

Continue to this website (not recommended).

More information

Websites with invalid and valid certificates, as shown by Microsoft Internet Explorer.



Website Identification

GeoTrust Global CA
has identified this site as:

www.google.co.uk

This connection to the server is encrypted.

Should I trust this site?

View certificates

In HTTPS, certificates play two very important roles: they prove the authenticity of the website, and they help the connecting parties establish a secure communication channel despite their communications possibly being "overheard" and even tampered with.

This is all backed by strong cryptography, so we can speak of undeniable authentication and a cryptographically secure communication channel. This means that it is mathematically impossible for an outsider to read data exchanged by the parties, even if the outsider has full access to their communication channel.

As cryptography used in TLS is very hard to defeat, a more realistic route for a prospective hacker to get into someone's TLS channel is based on bringing an intermediary proxy server into the scheme. Basically, an eavesdropper sets up his own TLS server, and somehow (through DNS spoofing or some other technique) makes the browser use the intermediary server instead of the intended one. When the proxy accepts a connection from the browser, it sets up its own TLS connection to the correct destination and simply relays all data received from the browser to the destination and back, while keeping record of it or altering it if needed.

There is one problem in this route, though. The strong cryptography behind digital certificates doesn't allow the proxy to use the certificate of the original TLS endpoint. That's why a typical next step for the eavesdropper would be to create a fake certificate, with all the information contained in it identical to that of the genuine web site, and attach it to their proxy. When the browser connects to the proxy, the intermediary will feed it the fake certificate, with the hope that the browser or the user won't notice the difference.

This is a real risk, which is mitigated by restricting the scope of authorities that can issue digital certificates to public entities.

Digital certificates for legitimate websites are issued by reputable trust providers (certification authorities, or CAs), such as Verisign, Thawte, or GlobalSign. Before issuing a certificate to a website, the provider performs background checks on its owners, making sure that the company behind the website is genuine and complies with all relevant regulations.

Every browser maintains a list of recognized trust providers, and only trusts certificates issued by the providers that are on that list. This prevents black hats from creating their own trust providers, and using those for issuing certificates for illegitimate or harmful websites, such as the mentioned proxy server. Any website presenting a certificate issued by such a provider will be immediately deemed untrusted by popular browsers, as the authority won't be on their trusted CA list, and they will show a warning similar to the one above and will suggest aborting the connection.

While not without drawbacks, this trust system has proved to work, and has helped establish security, confidentiality, and privacy over the web.

Now that we are clear about the role digital certificates perform in securing the web, let's get back to our particular PC and the antivirus software running on it. As we can conclude from the above, under normal circumstances no application other than the browser and the HTTPS server it connects to can access the protected data exchanged between them. That's because HTTPS was specifically designed to protect any traffic from anyone, however "bad" the traffic is or however good the intentions of the eavesdropper may be.

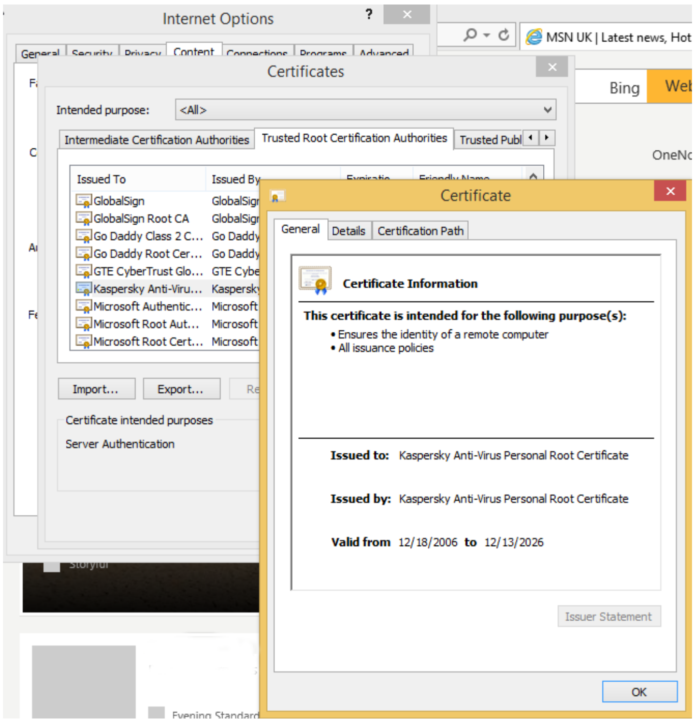
So how does KIS manage to get access to the protected traffic then? As I described above, it detects and intercepts the browser's requests for HTTPS resources (e.g. *https://www.facebook.com/*), and doesn't let them go any further than the local system. Instead, it quietly launches a HTTPS proxy server on the computer, and diverts the browser's requests to it.

The proxy server sets up its own connection to the resource in question, and then works as a switch, relaying data it receives from the browser to the HTTPS resource, and back.

Obviously, as it is an active participant in both connections, the tool can read and even alter all the "protected" data going back and forth. So that the browser does not to detect the presence of an alien HTTPS server en route, the antivirus tool generates a fake certificate matching the name of the HTTPS resource the browser is requesting, and attaches it to its proxy server. In the above example it would generate a certificate for *https://www.facebook.com*, which would keep the browser satisfied about the web site authenticity. But how is that possible – I've just said that the browser would detect a certificate by a CA that's not on its list of approved CAs?

As KIS has direct and full access to the computer's operating system, it can make changes to its configuration that would make the browsers comfortable with the certificate substitution. During its installation process, the tool quietly adds its own trust provider to the list of globally recognized trust providers maintained by the browser.

Once that's done, it can use that trust provider to issue any certificate it needs, and all such certificates will be fully trusted by the browser. That's because this trust provider has no identifiable differences from any other genuine trust provider residing in the system's trust list, and the browsers have, therefore, no reason to be suspicious of it. This little trick removes the last barrier preventing KIS from injecting itself into any secure connection without being detected and reported.



Intended purpose: <All>

Intermediate Certification Authorities | Trusted Root Certification Authorities | Trusted Publ

Issued To	Issued By
GlobalSign	GlobalSign
GlobalSign Root CA	GlobalSign
Go Daddy Class 2 C...	Go Daddy
Go Daddy Root Cer...	Go Daddy
GTE CyberTrust Glo...	GTE Cybe
Kaspersky Anti-Viru...	Kaspersky
Microsoft Authentic...	Microsoft
Microsoft Root Aut...	Microsoft
Microsoft Root Cert...	Microsoft

Certificate intended purposes
Server Authentication

Certificate Information

This certificate is intended for the following purpose(s):
- Ensures the identity of a remote computer
- All issuance policies

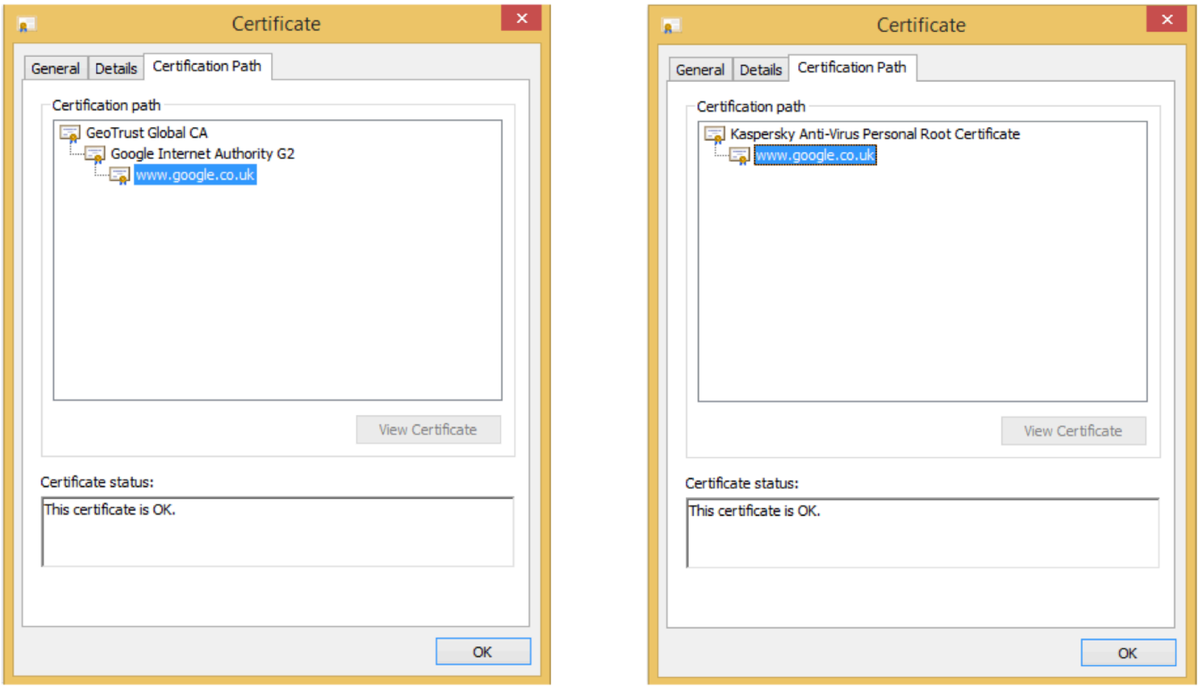
Issued to: Kaspersky Anti-Virus Personal Root Certificate

Issued by: Kaspersky Anti-Virus Personal Root Certificate

Valid from 12/18/2006 to 12/13/2026

Kaspersky pocket CA certificate in the system's trust list.

As the picture below shows, the identity of a secure website (*google.com* in this case) is then no longer confirmed by the original reputable authority, but by the “Kaspersky Anti-Virus Personal Root Certificate,” the trust provider residing on the PC.



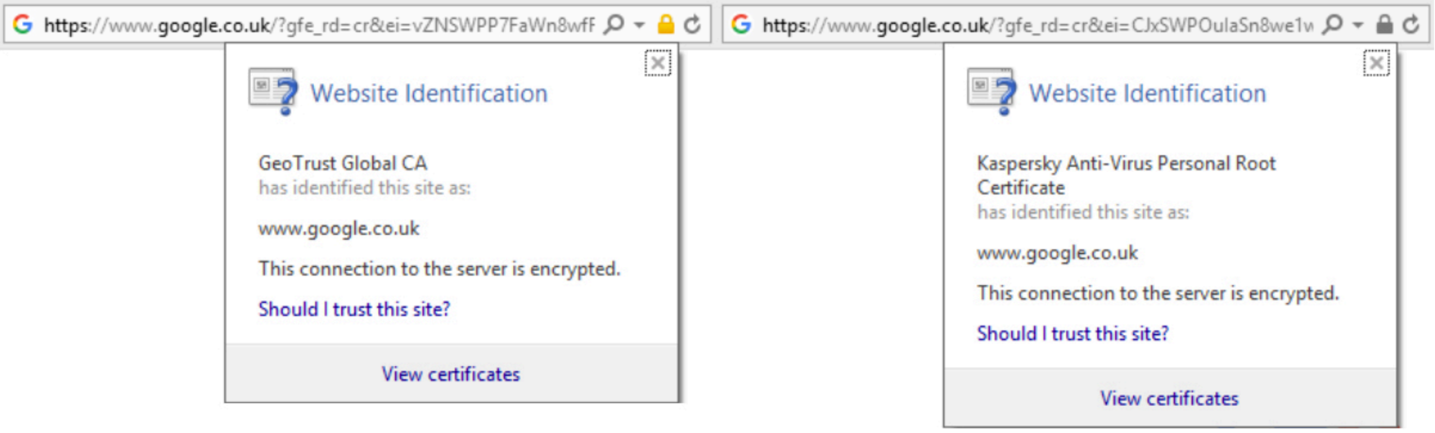
Certification path
GeoTrust Global CA
Google Internet Authority G2
www.google.co.uk

Certificate status:
This certificate is OK.

Certification path
Kaspersky Anti-Virus Personal Root Certificate
www.google.co.uk

Certificate status:
This certificate is OK.

Certificate chain for *google.co.uk* on a clean and a Kaspersky-protected computer.



https://www.google.co.uk/?gfe_rd=cr&ei=vZNSWPP7FaWn8wfF

Website Identification

GeoTrust Global CA
has identified this site as:

www.google.co.uk

This connection to the server is encrypted.

Should I trust this site?

View certificates

https://www.google.co.uk/?gfe_rd=cr&ei=CJxSWPOulaSn8we1w

Website Identification

Kaspersky Anti-Virus Personal Root Certificate
has identified this site as:

www.google.co.uk

This connection to the server is encrypted.

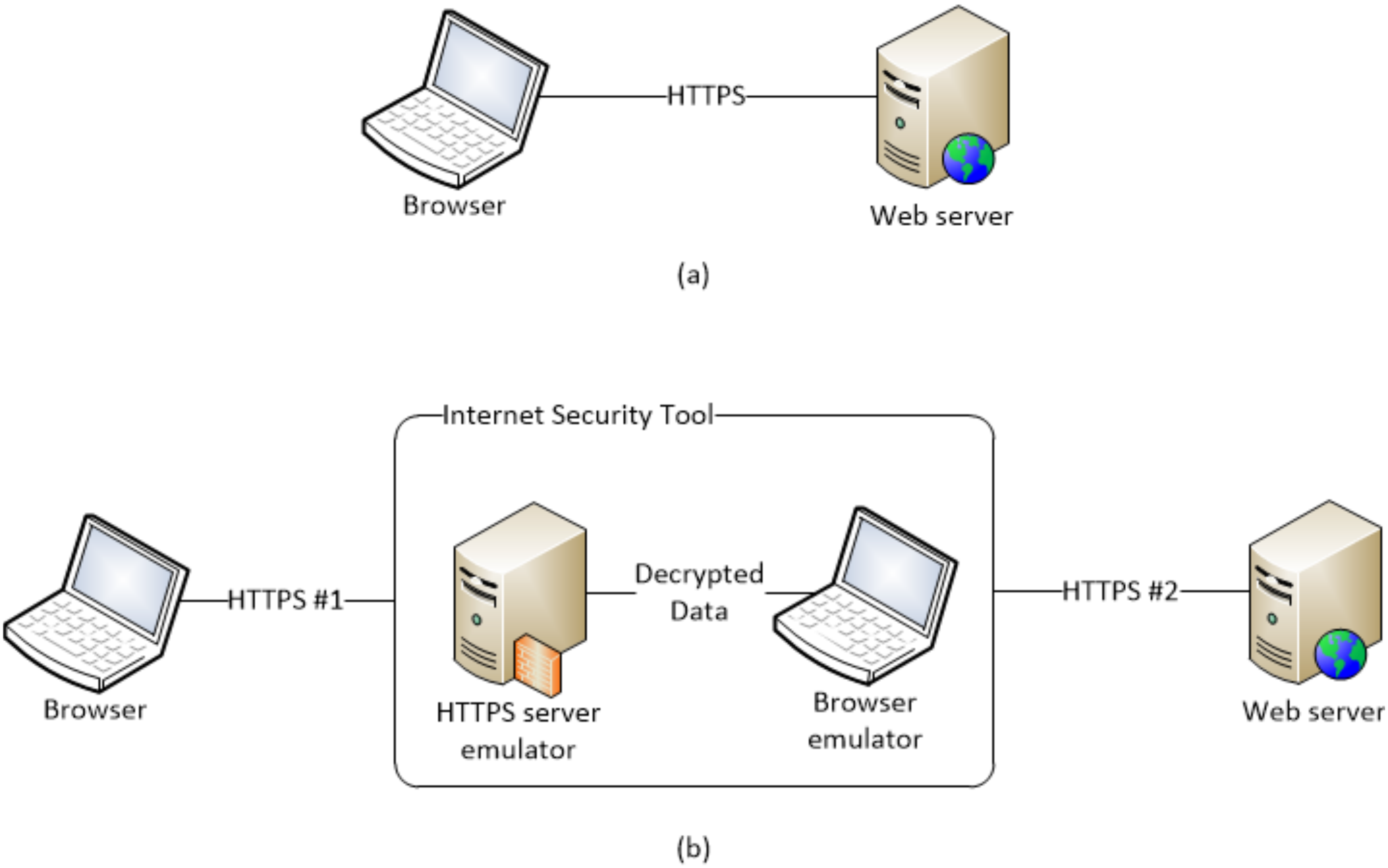
Should I trust this site?

View certificates

Security details for *google.co.uk* on a clean and a Kaspersky-protected computer (as shown by Microsoft Internet Explorer). Can you spot the difference?

Such activities can be qualified as what cryptographers define as an “active Man-in-the-Middle attack,” a term that refers to an adversary capable of monitoring and modifying data transmitted between A and B by presenting as B to A, and as A to B, without the deception being noticed.

My research has shown that KIS didn’t listen to all secure traffic originating from the computer. Still, the list of online resources to and from which secure traffic was intercepted is imposing, and features most of the popular online services, including Google, Facebook, Twitter, OneDrive, LinkedIn, Bing, and Amazon Web Services. The traffic to other secure websites (perhaps less known, or more static in nature?) is let through without interception.



(a) Original communication route, (b) altered communication route.

After sharing my findings and concerns with friends and doing some field research, I discovered that the KIS tool is not alone in using this questionable technique. At least one other tool, by Czech company Avast Software, uses a very similar certificate substitution approach to listen to encrypted traffic. This might be an indication of a growing trend on the anti-malware market, with practices once considered a strict no-no crawling their way in.

The implications of this practice

Straightforward information security techniques, such as strong cryptographic algorithms or fortified data centers, are not the only instruments providing security to our data travelling across the Internet. An equal (if not greater) contribution to security on the Internet is made by non-technical trust relationships between the parties involved. While you can easily measure the strength and resilience of an employed technical instrument (e.g. a cryptographic key), it is very difficult to measure the value brought in by a written or unwritten trust contract between the players (be they humans or machines).

No wonder that the vast majority of attacks is leveraged against the target's trust relationships rather than cryptographic algorithms (social engineering is nothing else but exploiting trust relationships in humans).

When it comes to HTTPS, there are two forms of trust relationships involved. The first form is covered by what is called the global Public Key Infrastructure (PKI). PKI is a very clever way to organize the whole changing variety of endpoints connected to the Internet and make the task of establishing trust to every particular entry straightforward and simple. Best of all, this is done without any prior knowledge of the entity. The chain-of-trust method allows your browser to immediately classify websites as trusted or untrusted, without the need to perform a fully-fledged due diligence process each time it connects to a new website. Instead, the due diligence process is delegated to dedicated certification authorities. All this results in uninterrupted delivery of consistent, valid, and up-to-date trust information to all browsers around the world.

Trust relationships of the second form are between the users and the vendors of their web browser and operating system, whom they trust to conscientiously implement any needed security mechanisms and to carefully select which certification authorities to trust on their behalf. The trust users have in major operating system vendors is often built on their history and reputation, while the trust in smaller browser vendors is often based on the open source nature of their code, which allows everyone to learn how they work and make sure they do all the things right.

Summing up, you trust certification authorities to perform their due diligence duties regarding secure websites, and you trust your browser to pick the right authorities and employ the right security algorithms for your secure connections.

Every piece of this trust ecosystem was carefully put in place over the course of years, responding to newly identified threats and polishing itself up to be tolerant to most common types of faults. This system might not be perfect, but it is definitely the best option we have today, and is a *de facto* and, progressively, *de jure* standard. Any change to this fragile ecosystem would inevitably result in discredit of the whole concept of Internet trust for all parties involved. A single player that stops following the rules, even if the majority of other players would prefer to continue sticking to them, corrupts the system, and would eventually bring all the trust relationships in the system to an end.

By registering an artificial, quasi-trusted certification authority in users' system, the anti-malware tools bypass all security mechanisms that are there to protect the users, and invalidate the whole concept of Internet trust for everything they chance to browse from their computer. Instead of trusting the Google's website through a verifiable public chain of trust (google.co.uk is trusted by Google Internet services CA → Google Internet Services CA is trusted by GeoTrust, Inc. → GeoTrust, Inc. is trusted and regulated by the US government), the browser is now made to trust it just because the local copy of Kaspersky Internet Security "told" it to do so through its certification authority.

In essence, the only fact that the user can be totally sure about now is that the Google website opened in his or her browser was deemed to be OK by the local copy of Kaspersky Internet Security. The user doesn't know anything else about the origins and the legitimacy of that website.

Besides the general trust issue, a security professional can point out the following selection of risks brought in by the anti-malware tool:

Surveillance. The mere thought that my secure traffic is being quietly listened to leaves a bad taste in my mouth. Whenever I see that *https://* prefix in the address bar, I expect the communication to remain solely between that website and me. There are personal emails that I don't want anyone to see, there is sensitive information that was shared with me by someone who trusts me and, after all, there is my payment card information that I want to keep secret and safe. All in all, HTTPS is there for a reason.

I don't like the idea of tools like KIS gaining direct access to all that data. I have no idea what data the tool gathers from my private communications and where it sends it. I know for sure that KIS does send something to KSN, its own cloud, which officially provides up-to-date "oracle" services to desktop installations of the tool. Still, it is not possible to establish the nature and scope of the data uploaded to KSN, as KIS uses a proprietary security protocol, not TLS, to protect all communications between itself and its cloud component.

That is strange, isn't it? And really worrying. For example, by intercepting my Google traffic, they – whoever that "they" may refer to – can gain complete access to all my Google accounts, including emails, photos, calendars, and files on my Google drive, all the while pretending to be me. They can see my login information in the clear (another reason to employ 2-factor authentication, which involves a separate authentication channel outside of your PC, yet it doesn't always help as the tool may simply intercept the authorization cookies). Even if they don't do that on behalf of their company, collecting data of that nature requires extraordinary responsibility and care to exclude even the smallest chance of it getting into the wrong hands.

What is even more worrying is that the use of a very similar concept of intercepting private traffic was broadly discussed by Russian officials and the country's federal security service last summer. Its resemblance to KIS's technique is so glaring that it can hardly be taken as a coincidence. Essentially, KIS employs exactly the model leaked by the security service's whistleblowers, with the only difference of residing on end users' computers and not at the Internet service providers' facilities.

Substitution and misrepresentation. The data intercepted in this way can be passively listened to, but can also be modified in every possible manner, without the user being able to spot any change. The assortment of possible changes to the content ranges from harmless insertion of ads by companies affiliated with Kaspersky Labs or activity-tracking JavaScript, to removal of harmful content (with what is and is not harmful content being defined by the tool).

This means that you can miss an email or a Facebook post if KIS considers it harmful. Ultimately, you never know what kinds of changes such software might be capable of introducing to your traffic. In theory, it can show you anything it wants, and you will never be able to tell if that content is genuine.

Trust compromise. As I mentioned before, trust relationships on the Internet are carefully managed through a selection of interconnected mechanisms, from cryptography-driven public key infrastructures and government regulations, to open browser architectures.

By embedding its own CA in between your browser and the Internet, tools like KIS or Avast Free Antivirus essentially destroy and discard the whole trust ecosystem on your particular computer, effectively cutting it off from the global trust environment, and replace it with their own personalized understanding of trust. In short, they tell you: "Forget everything you knew about trust, and let us decide. Don't trust your postal service, your bank, or your garage anymore. Just trust us, and we will sort out any trust relationships for you in our own way."

And this is no good. Under normal circumstances, the trust for the websites you are connecting to, be it a giant like *google.com* or a smaller *flowershopnebraska.com*, is guaranteed by the whole World Wide Web trust ecosystem, which involves a number of mechanisms for preventing fraud and misuse. The chance of a malicious website getting through that ecosystem and ending up being shown as trusted in your browser is incredibly small.

When you give a tool like KIS a monopoly to decide which website is trusted and which is not, you are taking on an enormous amount of risk. You can't be sure whether a website considered to be trusted by the tool is actually trusted in terms of the WWW trust ecosystem. Intentionally or due to some nasty coding error, the interceptor may alter the actual trust figures, making the browser display untrusted websites as trusted, and vice versa. Ultimately, combined with its capabilities of making changes to secure traffic, the tool may literally feed your browser anything it chooses to, and make it seem trusted!

Single point of failure. A failure of a particular link in the global trust ecosystem is an unpleasant but controllable event. As most of the players are prepared to handle various kinds of trust issues, most of those won't lead to big problems. Revocation of a certificate that got stolen takes just a few minutes, so a malicious website can be taken down nearly instantly, before causing too much harm, and in a fully automated way.

You can't be so sure about that when a third-party, closed-source tool is involved in controlling all aspects of your communications. Since all the traffic and all the trust relationships are now handled and controlled by that single application, that application becomes an easy target for attackers of all sorts, as it is become a single point of failure.

And what a point of failure it is! Really, the trust for the whole Internet now depends on a single CA certificate residing on your computer, a consumer level device that may or may not be secure. Just for the sake of comparison, the usual WWW trust scenario distributes the trust evenly between a few dozens of primary CAs and hundreds of smaller ones, each coming with extended protection measures like intrusion prevention systems, electrified fences, CCTV, and magnetic fields. Getting access to proper CA certificates requires enormous amounts of effort, which is statistically confirmed by the fact that the cases of CA certificate compromise are extremely rare. Conversely, recovering KIS's quasi-CA certificate from your PC is a very simple task, totally achievable through a small piece of malware slipping through your mail or a fresh bug in your browser. And once the attacker gets access to that certificate, they can quietly join KIS in its surveillance program.

It is also worth noting that running a local TLS server for every outgoing connection imposes a significant burden on the computational resources of your computer. While web servers are tailored to handle loads of incoming requests, personal computers are not as powerful, and can be easily overloaded by a few dozens TLS requests, which doesn't benefit the stability of your local system and might make it more open to attacks.

Finally, everyone is a human, and it's impossible to guarantee that KIS or Avast's solution are free of programming errors and bugs. Some of them may let attackers into the system, while others may simply make the application misbehave when establishing trust to a particular website. I'm not saying that browsers or CA software are free of bugs, but the levels of quality control employed at Kaspersky Labs and a CA like Verisign are incomparable, and a personal anti-virus tool is a much more attractive target for attackers than a protected CA.

Unknown rules. The Internet has always been a very liberal, very open place, welcoming those contributors playing fairly and transparently. The internal details of nearly every protocol used within the Internet infrastructure have always been open to the public, and often contributed by the public. The open model appeared to be quite efficient with regards to the overall quality of the underlying technologies, and particularly with regards to the quality of security protocols, since everyone interested was able to have a look inside, assess the employed techniques, and draw attention to any recognized flaws.

Conversely, the design of KIS raises more questions than answers. The software's source code is proprietary, and only Kaspersky Labs employees know what it does and how it does it.

We know that while in operation, the software sends something to KSN. Again, there is no way to check what exactly is being sent, as the tool uses a proprietary protected protocol to encrypt all outgoing data.

It is also unclear why the software treats different websites differently. Does it consider Twitter more risky than *flowershopnebraska.com*? If I was an attacker and wanted to distribute malware from a secure website, I would probably consider achieving clearance from a CA for a small web site, and serving malware from it.

We also don't know what else this software is capable of beyond what is written in its tech specifications. Recent political tensions and accusations of hacking suggest that there might be something not addressed in the user manual. A lot of sensitive information was stolen and revealed in recent months. Can we be sure that the information was actually stolen through *illegal* hacking and not *legal* hacking – namely, by some anti-malware solution, quietly making its way through all the mail traffic on the victim's PC?

Conclusion

The trend of employing controversial techniques to gain access to secure web traffic as adopted by anti-malware software from various vendors is quite worrying. TLS and the whole certificate infrastructure are there to protect sensitive information by guaranteeing its confidentiality, authenticity, and integrity, with the little green lock in the address bar of your browser becoming a symbol of privacy and trust.

Even though an Internet security application is technically capable of taking a (rather questionable) shortcut to circumvent the TLS layer and gain access to the traffic it protects, the mere availability of this shortcut should not be perceived by the vendor as a call to action.

The method employed by Kaspersky Internet Security and other tools to get to the protected data should not be acceptable. By intruding into the trust chain, these tools effectively remove all benefits offered by the Internet trust infrastructure, and create a wide scale single point of failure. Ironically, by doing so they discard all security services TLS offers to users.

There is no confidentiality as users' data is being constantly overseen, no authenticity as there is no original trust chain, and no integrity as there is no way to identify whether the security tool made any changes to the data. Effectively, the security tool removes all security from one of the most widely used security protocol!

Indeed, while listening to secure traffic may have some objective benefits, the method chosen by Kaspersky and Avast is far from being reasonable. Even if we assume that there is no global surveillance conspiracy behind it (and I fully admit that's possible), there are just too many risks in doing it that way.

Monitoring traffic at the ends of the secure pipeline as it enters and leaves it would provide the same output for the anti-malware tools but without affecting the benefits offered by the TLS service. In fact, it would only add to the overall security of the user's environment, making all security components work together in symbiosis and harmony.

Finally, such an unscrupulous and rough neglect of common security practices and manipulation of the global trust may eventually lead to the compromise of confidence in HTTPS, TLS and other security protocols. If it's acceptable for KIS to use such methods to intercept secure traffic, why wouldn't it be for any other security tool? Really, KIS might be only the last player in the chain of listeners to your HTTPS traffic employing the same methods, with unknown number of others sitting in between your PC and the web servers. With our data being watched and trust relationships being distorted at every link of this hypothetical chain, how can we believe in the privacy, security, and authenticity of our data at all?

Ken Ivanov, PhD, is the Director & Chief Security Expert at EldoS Corporation (www.eldos.com).

Evolving PKI for the Internet of Things

By Jeremy Rowley

The rapid growth of the Internet of Things is outpacing security implementations, and the industry desperately needs to stem the tide of risks that come with it.

IDC estimates that, by 2020, the number of Internet-connected devices will surge past 200 billion. The sheer scale of this future Internet of Things means that it needs a strong security layer that is scalable, reliable and can be automated to meet the needs of a rapidly growing market.

Cryptography is one solution that can provide a strong security layer, with encryption and identity, at such a scale. And now, more than ever, security teams are looking to evolve public key infrastructure (PKI) to meet the challenges of IoT security.

A trusted security layer

Through the use of PKI, it's possible to achieve many needed security functions within the IoT, such as:

Device authentication: PKI can help establish mutual authentication for all connections and provide access control to ensure only authorized parties can use the device and view its communications.

Data encryption: PKI ensures the encryption of data in transit, at rest, and in process.

Data and system integrity: PKI helps to validate the integrity of the data coming to and from the device (makes sure that the data has not been manipulated in transit). It can also help with secure device boot, configuration settings, and IP protection. Certificates can help protect device patch management by verifying the code and authenticating it to the proper device.

Achieving all this, however, requires automation and ingenuity.

Evolving PKI for the IoT: A matter of scale

Traditionally, security professionals associate PKI with manual, time-consuming and complex processes that are required to procure and install an X.509 digital certificate onto a physical server, one at a time.

These processes are improving for physical server authentication and encryption, but when trying to imagine placing one certificate at a time onto IoT devices, security teams grow weary. The sheer scale of the IoT makes traditional methods of managing PKI unreasonable.

Security administrators neither have the time nor interest in deploying connected device security in this way. Evolving methods of automating PKI make it a leading technology for providing the needed identity and encryption for IoT device communication, but some challenges and misperceptions still exist. These include:

- Balancing public certificate lifetime standards with device lifecycles
- Provisioning certificates at a scale
- Managing certificate lifecycle events
- Running a secure and compliant CA, and
- Securing low-compute devices that can handle very little data.

And all of this has to be done within a budget.

Public vs. private trust: IoT device use varies wildly, and some organizations with public-facing applications require public trust. This means that companies need to think about the certificate lifetimes and other requirements established by public trust stores, and ensure that certificates can be updated as these requirements change. For example, the CA/Browser Forum recently shortened the maximum validity period of certificates to 27 months. Manufacturers of devices incapable of renewing certificates within that period should carefully consider whether public trust is required.

In many cases, IoT devices communicate within closed systems, meaning trust within the Web PKI ecosystem isn’t required. Avoiding public trust offers the device manufacturer greater flexibility in how the device uses PKI, including managing device lifetimes. Running a private PKI is no small matter, and companies should consider the complexities and costs of running their own systems versus hiring an experienced third-party.

The sheer scale of the IoT makes traditional methods of managing PKI unreasonable.

Provisioning credentials/certificates: With so many devices to keep track of, companies face the challenge of provisioning these devices manually. At the same time, many IoT devices are not built for Internet connectivity or do not include a thoughtful security design.

Often, devices fail to follow one of the common open certificate enrollment standards such as SCEP or EST. For those that do, certificate provisioning can be greatly simplified through the use of APIs that plug into companies’ systems and handle large-scale issuance of millions of certificates at a time. Cisco and DigiCert recently worked on a successful testing program to demonstrate how EST can be used for certificate enrollment on IoT devices.

Certificate footprint: Another factor is the computer power and processing capabilities of the devices. Some of these devices may not have the processing capacity to run high-footprint certificate services. But thanks to recent innovations, there are now low-power computing solutions that any device can use. The challenge lies in helping manufacturers understand these technology advancements and providing a mechanism for updates during the device lifecycle.

One of the biggest PKI costs is the equipment and expert personnel necessary to stand up an internal CA.

We can look to NFC devices for an example of how the low-compute challenge can be overcome. With little power or no power source at all, NFC devices can actually use PKI to protect content from modification or deletion. Although some of these devices may require unique key provisioning situations, they can still be fitted to use PKI throughout the device lifecycle. In addition to advances in TLS libraries, the certificates themselves have evolved to become lightweight vehicles that provide identity and encryption.

Certificate profiles have gotten smaller with new key generation methods and unique non-X.509-based formats.

Digital certificate lifecycle management: Digital certificate lifecycle management can be challenging in the enterprise, and the problem gets bigger with the IoT. Companies need to learn to manage their certificates by exception (to identify problems areas and address them) and find ways to use policy-based management tools. Setting up proper discovery, revocation, and renewal systems requires thoughtful planning.

Managing a PKI ecosystem is not easy. Thus, many companies look to an expert third-party, such as a publicly trusted CA, that has many years of experience complying with industry standards, maintaining the large infrastructure and security requirements necessary for a trustworthy CA, and setting the policies and practices. If companies opt for a private PKI system, they may also consult a CA to help establish their infrastructure.

Cost: Though people often think of PKI for the IoT in the same way they think about buying certificates for their web servers (one certificate at a time), the economies of scale of the IoT help drastically reduce costs and make bundling quite cost-effective.

One of the biggest PKI costs is the equipment and expert personnel necessary to stand up an internal CA. There are often hidden infrastructure costs such as the load on HSMs to isolate keys, which could be otherwise used for other data needs. Depending on the size of a company's deployments, they may be required to purchase separate HSMs for their PKI, as well as other parts of the security stack such as reliable revocation systems and signing servers. The net result is an unexpected operational cost that can leave a project unprofitable or unoptimized.

In many cases, the costs tied to the managing of the equipment, policies and infrastructure for an on-premises CA are much higher than outsourcing the task to a trusted third-party cloud provider.

Engage an expert

Deploying security onto IoT devices is not easy, but many companies are already moving towards deploying PKI-based solutions. The first step is finding a partner that is well versed in this work, so he can help you assess your needs, and determine whether PKI can work for your IoT deployments.

Jeremy Rowley is the VP of Emerging Markets at DigiCert (www.digicert.com).

Malware world

Ransomware spiked 752% in new families

2016 was truly the year of online extortion. Cyber threats reached an all-time high, with ransomware and Business Email Compromise (BEC) scams gaining increased popularity among cybercriminals looking to extort enterprises. A 752% increase in new ransomware families ultimately resulted in $1 billion in losses for enterprises worldwide, according to Trend Micro.

Trend Micro and the Zero Day Initiative (ZDI) discovered 765 vulnerabilities in 2016. Of these, 678 were brought to ZDI through their bug bounty program, then ZDI verifies and discloses the issue to the affected vendor. Compared to vulnerabilities discovered by Trend Micro and ZDI in 2015, Apple saw a 145 percent increase in vulnerabilities, while Microsoft bugs decreased by 47%.

The use of new vulnerabilities in exploit kits dropped by 71%, which is partially due to the arrest of the threat actors behind Angler that took place in June 2016.

“As threats have diversified and grown in sophistication, cybercriminals have moved on from primarily targeting individuals to focusing on where the money is: enterprises,” said Ed Cabrera, chief cybersecurity officer for Trend Micro. “Throughout 2016 we witnessed threat actors extort companies and organizations for the sake of profitability and we don’t anticipate this trend slowing down. This research aims to educate enterprises on the threat tactics actively being used to compromise their data, and help companies adopt strategies to stay one step ahead and protect against potential attacks.”

In 2016, the Trend Micro Smart Protection Network blocked more than 81 billion threats for the entire year, which is a 56% increase from 2015. In the second half of 2016, more than 3,000 attacks per second were blocked for customers. During this time, 75 billion of blocked attempts were email based.

How the Necurs botnet influences the stock market

The Necurs botnet is back to flinging spam emails left and right. It now distributes emails with no malicious attachment or link.

According to Cisco Talost researchers, the botnet has been spotted firing off short-lasting but sizeable bursts of penny stock pump-and-dump emails. The messages tout InCapta Inc., a mobile application development company, as a company with revolutionary drone technology, and say that it is going to be bought out at $1.37 per share by drone company DJI next week. Recipients are urged to buy its stock now, at 20 or less cents per share, and then sell it to DJI next week and make a killing Most people will be sceptical about those claims, but this type of spamming effort works.

"The stock has seen a significant increase in the volume of shares being traded," the researchers noted. "While analyzing this particular spam campaign, we observed that the volume of shares being traded reached over 1 million shares (the total later in the day was over 4.5 million shares), which is exponentially higher than the average volume of shares traded."

A second wave of very similar emails, sent eight hours after the first one, again increased the stock price.

This is not the first time that the Necurs botnet has been spotted fuelling pump-and-dump stock scams. The researchers pointed out a similar campaign in December 2016, when recipients were urged to buy stocks of a mobile application development services company.

Fileless attack framework was used in many recent attacks

Recently, a number of security companies spotted attackers targeting a variety of organizations around the world with spear-phishing emails delivering PowerShell backdoors (some of them fileless), misusing legitimate utilities, and communicating with C&C servers through DNS traffic.

In February, Kaspersky Lab researchers said the targets were mostly banks, and that the initial infection vector was unknown. Then, in March, Cisco Talos researchers detailed how the backdoor RAT used by the attackers uses DNS TXT message requests to talk to the C&C server, and FireEye said that they detected similar attacks targeting employees of US-based businesses that are in charge of filing reports with the US SEC. Now Morphisec researchers say that the three attacks were likely performed by the same criminal group, by using a sophisticated fileless attack framework.

"Initial infection begins when the weaponized Word document delivers a PowerShell agent that opens a backdoor and establishes persistency. After this point, in most cases, the rest of the PowerShell commands are delivered through the command server," Morphisec researchers summarized the attack.

"For some targets, the attack was fully fileless, eventually delivering a Meterpreter session directly to memory. In other cases, the password-stealer LaZagne Project or another Python executable was delivered and executed. After additional investigation, we identified controllers for different protocols including Cmd, Lazagne, Mimikatz and more."

The researchers also got to (for a brief moment) interact with the attacker via the PowerShell protocol used for the attack delivery, and poke around one of the C&C servers. And even though the attacker soon after that blocked one of the researchers' IPs and shut down that particular C&C server, the researchers managed to gain some insight into the setup.

"We found and downloaded a set of malicious files, some of them well-known and used for Mimikatz attacks, others are PowerShell exploitations and User Account Control (UAC) exploitations," they noted, and added that their brief interaction with the threat actor "made clear that the hacker is part of a group which limits their exposure by targeting specific companies only."

Intel's CHIPSEC can detect CIA's OS X rootkit

As details about CIA's hacking capabilities and tools are, bit by bit, rising to the surface, companies are trying to offer users some piece of mind. In the wake of WikiLeaks' release of the CIA document dump, Apple has stated that many of the revealed iOS exploits have already been patched, and the company is constantly working to address any new vulnerabilities. But it was Intel Security that offered a tool that can identify an EFI (Extensible Firmware Interface) rootkit that is meant to function as a covert implant on machines running OS X.

The rootkit is named DarkMatter, and is part of the DerStarke bundle for targeting OS X machines.

"[DarkMatter] appears to include multiple EFI executable components that it injects into the EFI firmware on a target system at different stages of infection," Intel Security's Christiaan Beek and Raj Samani explained.

"If one has generated a whitelist of known good EFI executables from the firmware image beforehand, then running the new tools.uefi.whitelist module on a system with EFI firmware infected by the DarkMatter persistent implant would likely result in a detection of these extra binaries added to the firmware by the rootkit."

The original firmware can be provided by the manufacturer (Apple).

The module they mention is part of Intel's CHIPSEC open source framework for assessing the security of a variety of personal computer platforms (hardware, system firmware, and platform components). CHIPSEC can be run on Windows, Linux, Mac OS X and UEFI shell, and includes a security test suite, tools for accessing various low level interfaces, and forensic capabilities.

Android devices delivered to employees with pre-installed malware

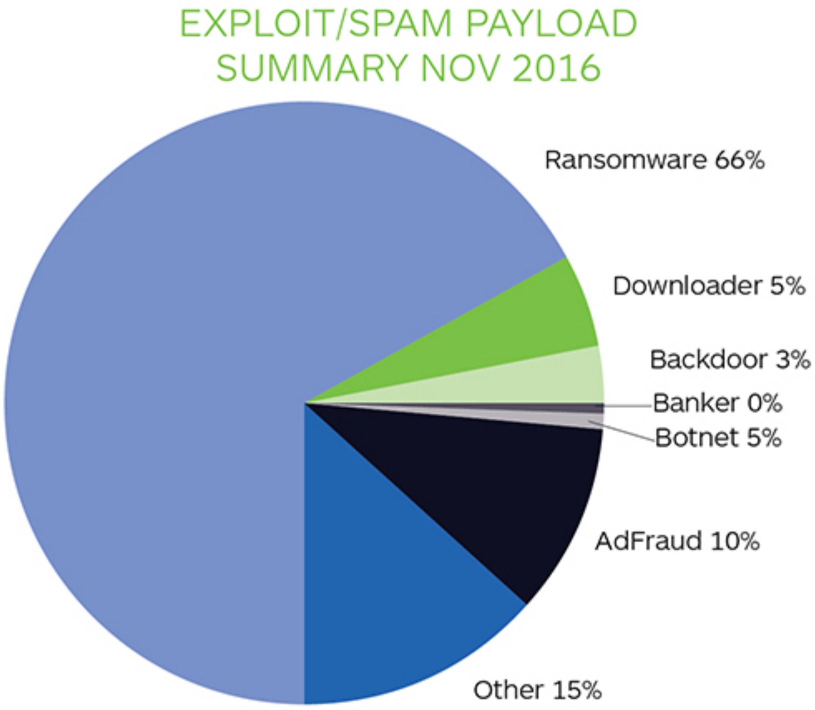
A test of Android devices used in two unnamed companies revealed that 38 of them were infected with malware before being delivered to the employees. These were smartphones by Samsung, ZTE, Oppo, Asus, Lenovo, and Xiaomi, but the manufacturers are not to blame for the malware. Check Point's research team was able to determine when the manufacturer finished installing the system applications on the device, when the malware was installed, and when the user first received the device.

They concluded that the malware were added somewhere along the supply chain, but could obviously not pinpoint when it happened. They also did not name the organizations to which the devices belonged – they just noted that it was "a large telecommunications company and a multinational technology company."

Top phishing targets in 2016? Google, Yahoo, and Apple

For every new phishing URL impersonating a financial institution, there were more than seven impersonating technology companies. Data collected throughout 2016 by Webroot clearly demonstrates a significant change since 2015, when the ratio was less than one to three. This increase may indicate that it is easier to phish a technology account, and that due to password reuse, they can be more valuable to hackers as a gateway to other accounts. The top three phishing targets in 2016 were Google, Yahoo, and Apple.

Researchers also uncovered a decreasing lifecycle in phishing attacks. The longest-running phishing site was active less than two days, and the shortest was only 15 minutes. Eighty-four percent of all phishing sites were active less than 24 hours.



The emergence of new global cybercriminal attack patterns

The findings of a new Malwarebytes report illustrate a significant shift in cybercriminal attack and malware methodology from previous years. Ransomware, ad fraud and botnets, the subject of so much unjustified hype over previous years, surged to measurable prominence in 2016 and evolved immensely. Cybercriminals migrated to these methodologies en masse, impacting everyone.

To better understand just how drastically the threat landscape evolved in 2016, researchers examined data taken from Windows and Android devices running Malwarebytes in more than 200 countries. Both corporate and consumer environments were studied and data was collected from June 2016 through November 2016. In the six months studied, nearly 1 billion total malware detections/incidences were reported. Data was also obtained from Malwarebytes' internal honeypots and collection efforts to identify malware distribution, not only infection.

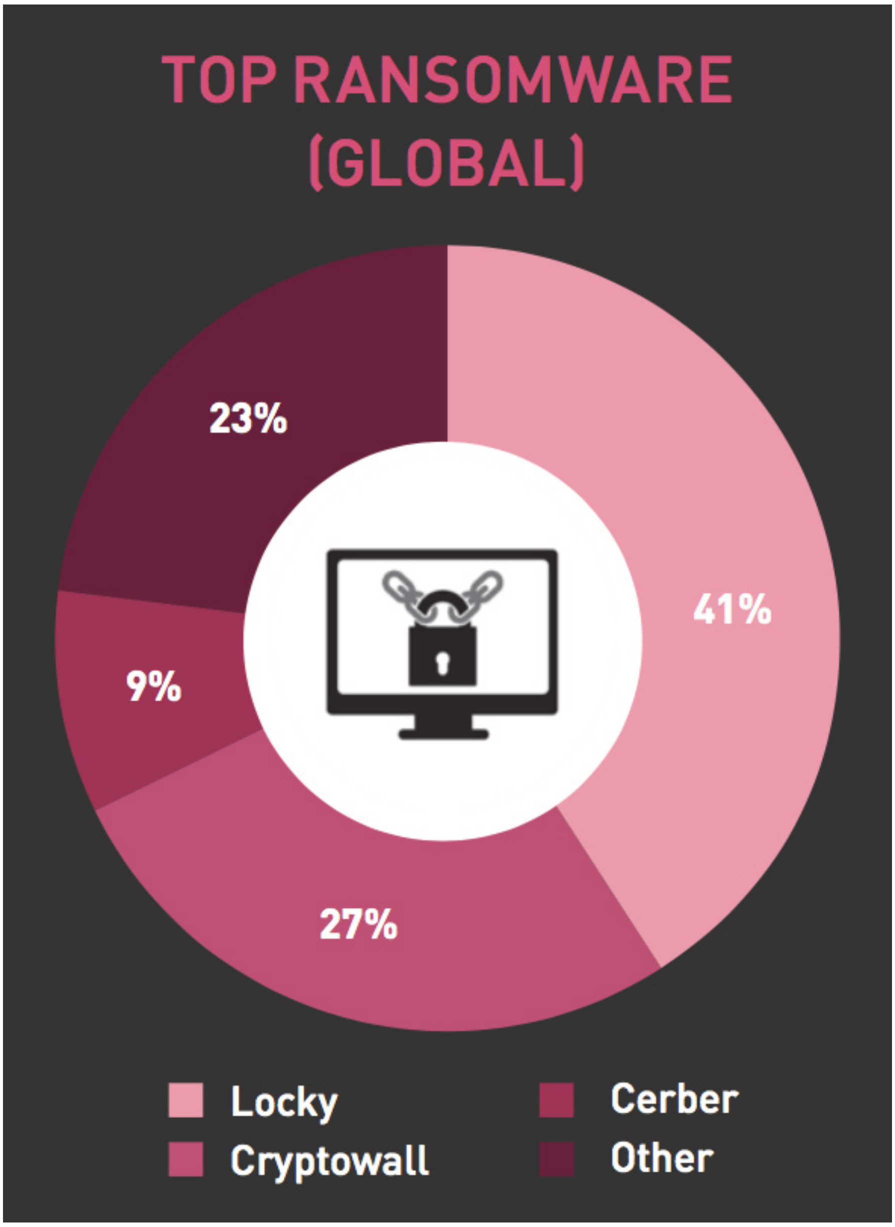
"To protect users from cybercriminals, we need to intimately understand their methodologies and tactics," said Marcin Kleczynski, Malwarebytes CEO. "Our findings demonstrate that the frequency and variety of new cyberattacks has crashed into people and businesses at an alarming rate. The last year involved an onslaught of ransomware, a surge of pernicious ad fraud and new, dangerous uses for botnets. These threats have the potential to erode many of the gains that computing is providing global society. Both consumers and businesses need to better understand how these new attack methodologies may impact them."

Ransomware attacks growing rapidly, organizations are struggling

The percentage of ransomware attacks increased from 5.5%, to 10.5% of all recognized malware attacks from July to December 2016, according to Check Point.

Check Point researchers detected a number of key trends during the period:

- The Monopoly in the Ransomware Market – thousands of new ransomware variants were observed in 2016, and in recent months we witnessed a change in the ransomware landscape as it became more and more centralized, with a few significant malware families dominating the landscape.
- DDoS Attacks via IoT Devices – in August 2016, the infamous Mirai Botnet was discovered – a first of its kind- the Internet-of-Things (IoT) Botnet, which attacks vulnerable Internet-enabled digital such as video recorders (DVR) and surveillance cameras (CCTV). It turns them into bots, using the compromised devices to launch multiple high-volume Distributed Denial of Service (DDoS) attacks. It is now clear that vulnerable IoT devices are in use in almost every home, and massive DDoS attacks that are based on such will persist.
- New File Extensions Used in Spam Campaigns – the most prevalent infection vector used in malicious spam campaigns throughout the second half 2016 was downloaders based on Windows Script engine (WScript). Downloaders written in Javascript (JS) and VBScript (VBS) dominated the mal-spam distribution field, together with similar yet less familiar formats such as JSE, WSF, and VBE.



7 real-world steps to security nirvana

By Bob Janssen

An organization pursuing its business goals cannot possibly dedicate 100% of its time and effort to achieving cybersecurity perfection, especially when you take into consideration the critical lack of cybersecurity talent that many firms are dealing with.

IT departments and cybersecurity pros are tasked with minimizing spending, empowering employees to be productive, and responding quickly to ever-changing security threats.

Below is a checklist of seven concrete steps that security professionals can take to protect their organizations without slowing down regular business operations:

Step 1: Reexamine and step up whitelisting policies

Ask: "Do we have a central repository of well-defined whitelisting policies?"

Dynamic whitelisting is one of the best ways to enforce access policies. It entails restricting user access and code execution by default to only that which is specifically permitted and known to be safe. Whitelisting should also take into consideration both identity and context attributes such as time of day, location or device. This model is essential for protecting your organization from all kinds of threats, including malicious hosts, hijacked user IDs, and insider threats.

A primary security requirement for an organization is, therefore, a unified repository of clearly defined whitelisting policies. These policies can be owned and controlled by different individuals with appropriate authority across an organization, but a single, reliable, and up-to-date place for maintaining whitelisting policies is essential across all resources, parameters, and user groups.

Step 2: Don't depend on "script heroes"

Ask: "Does our implementation and enforcement of our access policies still depend on manual configuration and/or homegrown scripts?"

Policies alone do not make a secure enterprise. An organization also needs a way to implement and enforce those policies in an automated way. Chances are, however, that an organization still depends on a wide range of disparate mechanisms to give users whitelist-appropriate access to digital resources. These likely include application- and database-specific admin tools and homegrown provisioning scripts.

There are many problems inherent in depending on these fragmented access provisioning mechanisms. From a security perspective, they are simply too unreliable because they are subject to human error and they're not intrinsically linked to the underlying policies they have been created to enforce.

If an organization still depends on "script heroes" to ensure the right people get access to the right resources at the right time, it is exposing itself to unnecessary risk.

Maintaining a unified, manageable, and automated mechanism for executing an organization's access policies can offset these concerns.

Step 3: When employees leave, make sure your data doesn't leave with them

Ask: "When someone leaves our company, are all of their digital privileges immediately, automatically, and entirely revoked?"

One of the single most important policy imperatives is the complete revocation of an employee's digital privileges immediately upon termination. Most organizations don't have a simple, automated, and reliable means of immediately eliminating an individual's access privileges across every application, database, SharePoint instance, communications service, etc. Some of those privileges can remain in place days, weeks, or even months after an employee is terminated, leaving the company exposed to risks that their breach detection and prevention tools can't do nothing about.

This is why in addition to having a unified system for managing access privileges across the enterprise, an organization also needs to appropriately integrate that system with whatever other systems can generate a valid termination event — including an organization's core identity management systems, HR applications, and contractor databases. Only such an integration can give an organization full confidence in the timely and complete revocation of digital privileges.

Step 4: Put access controls in place

Ask: "Can we reliably prevent users from accessing the wrong files from the wrong place at the wrong time?"

Most organizations can only apply a limited and relatively crude set of parameters to their access controls. In the real world, an organization's access policy parameters and controls must be richer and more context-aware. Common examples of this include:

• **Geo-fencing.** It often makes sense to constrain a user's access privileges based on location. A doctor, for example, may be allowed wireless access to certain clinical systems data while on premise at a healthcare facility, but not while off-site.

• **Wi-Fi security.** There may be times when an organization wants to make its data access rules (including read/write vs. read-only privileges) contingent upon whether a user's Wi-Fi connection is public/non-secure or private/secure.

• **File hashing.** File hashes provide a reliable means of ensuring that users only download, open, and work with legitimate content — thereby protecting an organization from a wide range of threats, including ransomware and spear-phishing attacks.

To implement these kinds of rich security controls, an organization needs an access management system that can automatically respond in real time to session context and execute hash-based identification.

Without those controls, defense against various types of identity and content spoofing will be severely limited.

Step 5: Make sure your security process is adaptable

Ask: "Do we have a consistent process for adding new applications (including cloud/SaaS) to our whitelist as demanded by the business, and for applying the appropriate policies to them?"

An organization's business isn't static. In fact, most companies are adding new cloud/SaaS services at a faster pace than ever. Many of these new services are being activated directly by lines of business, without much involvement from IT. At one time, this was referred to as "shadow IT." But it's not just a shadow anymore. It's central to how organizations leverage software and analytic innovation in the cloud.

If an organization can't secure these new applications and services quickly, there can be several unacceptable outcomes:

- Employees may be unable to use new resources in a timely manner because they're blocked by an organization's whitelisting system
- New resources may get whitelisted too hastily, without being properly secured by policies such as geo-fencing and Wi-Fi restrictions
- Worse yet, employees may come up with workarounds to avoid or bypass an organization's security mechanisms.

To avoid these outcomes, organizations need a fast, reliable, and consistent process for adding new cloud resources and conventionally developed applications to its whitelisting repository/automation engine. Without such a process, an organization's security won't be able to keep up with the business, and this means it will either compromise the former or impede the latter.

Step 6: Empower self-servicing

Ask: "Have we met the needs of the business for consumerization/self-service and LOB delegation?"

The millennial workforce is increasingly expecting IT to provide consumerized self-service similar to what they experience in their personal use of technology. Self-service is a win-win for IT and the business. The business wins because self-service takes delay out of everyday requests for digital services. IT wins because it frees staff with limited time from a variety of routine tasks. Self-service can also include the delegation of certain administrative tasks such as the authorizing access privileges or adding software licenses to line-of-business managers.

The best way to provide self-service and delegation to the business is by extending an organization's security whitelist automation engine to non-IT users with the appropriate policy-based controls. This approach allows an organization to make sure that no one outside of its cybersecurity team can violate its policies, and empowers those employees to quickly perform routine tasks without the IT department's intervention.

Step 7: Prepare for an audit

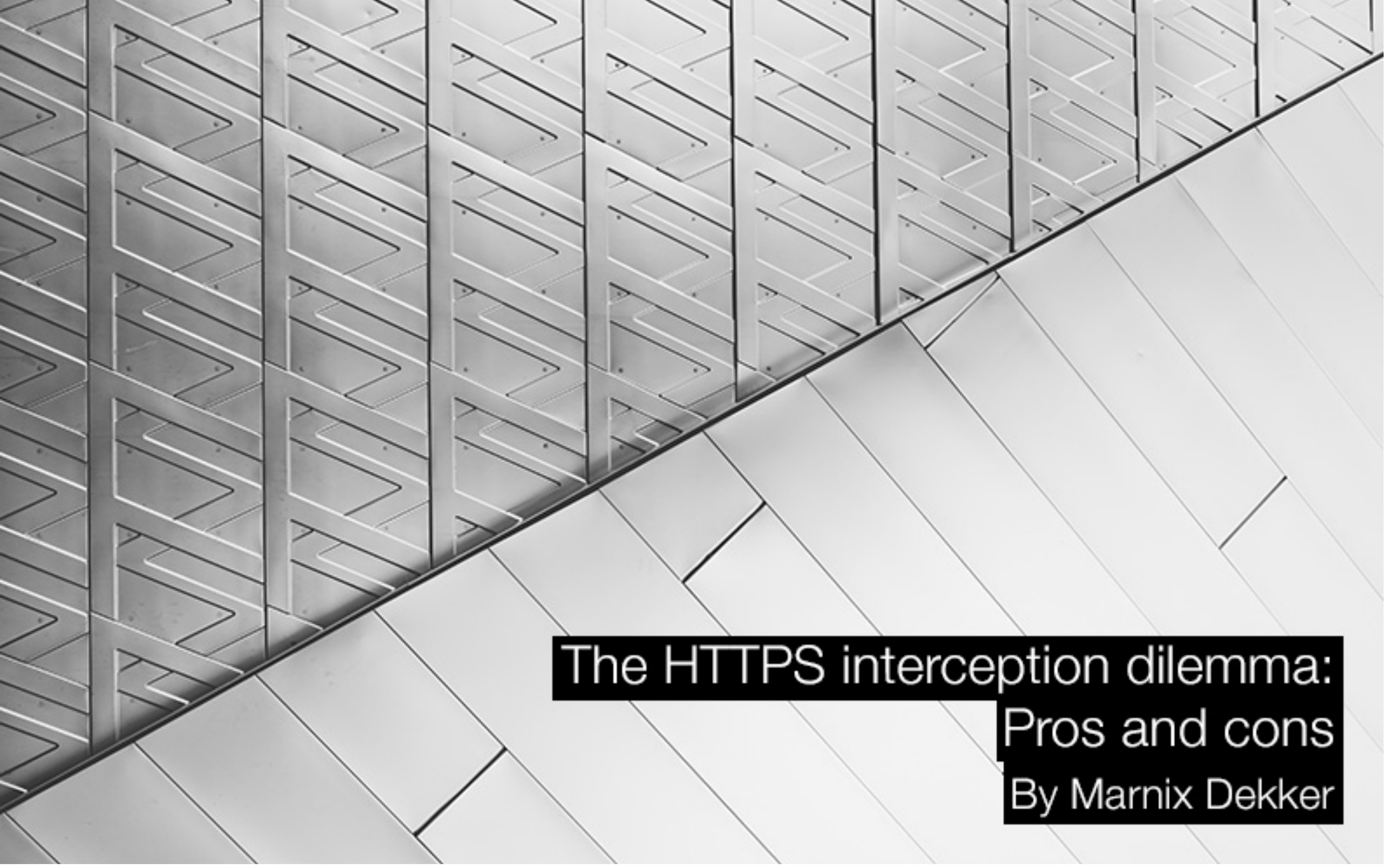
Ask: "Are we ready to handle an audit?"

An organization might implement all of the above noted six steps, but none of it will matter if they cannot credibly prove it to an auditor.

That's why an organization needs a unified, rules-based access whitelisting automation engine that's fully self-documenting. Only a centralized permissions control "brain" can secure an organization's environment and enable an organization to quickly and easily provide auditors with credible evidence that it has exercised full diligence.

By leveraging a single, robust access provisioning mechanism across all of its digital resources — from its most complex core business applications to its most recently adopted cloud service — an organization can make itself vastly more secure, while at the same time not adding unnecessary strain to the daily workload.

Bob Janssen is the CTO, founder, and SVP of Innovation of RES (www.res.com). During his tenure, RES has sold millions of licenses worldwide. Mr. Janssen holds several patents for the solutions he has developed at RES, and has worked with the RES R&D team on the filing of numerous others.



The HTTPS interception dilemma: Pros and cons

By Marnix Dekker

HTTPS is the bread-and-butter of online security. Strong cryptography that works on all devices without complicating things for users. Thanks to innovative projects like Let's Encrypt, adoption of HTTPS is rising steadily: in mid-2015 it was at 39%, now it's at 51%.

Recent research shows, however, that HTTPS interception happens quite often. In fact, about 10% of connections to CloudFlare are intercepted, and the main culprits are enterprise network monitoring products. Without HTTPS interception, a network monitoring tool will only "see" the Internet domain names and/or the IP addresses of the two sides of the connection, and not the full URL or the content of the communication.

But HTTPS interception is controversial in the IT security community. There are two sides in this debate. Much depends on the setting you are in. In this article I want to outline the benefits and drawbacks of HTTPS interception, from an IT security perspective.

The benefits of HTTPS interception

Detect malware downloads: In most cyberattacks the end-user is tricked into downloading malware or an infected file, for example via a phishing, a watering hole, or a malvertisement attack. HTTPS interception allows the network proxy to see the downloaded binaries and documents, and this means they can be scanned and compared with known malware signatures, or opened in sandboxes.

Detect C&C traffic: Command & Control traffic to exotic domain names and IP addresses is the hallmark of an infected device calling back to the attacker's infrastructure. To avoid detection, attackers have started to use legitimate websites for C&C traffic, for example a Twitter feed of a burner Twitter account. With HTTPS, you can only see that there is an Internet connection with Twitter, and it blends in with normal user traffic. HTTPS interception allows the Internet proxy to also see the content, for example which Twitter accounts are accessed. This approach could, in principle, be used to distinguish C&C traffic from normal user traffic.

Detect exfiltration: Attackers can use HTTPS connections to exfiltrate data. HTTPS interception can be used to detect if corporate documents or files are being uploaded to a remote server by looking for known patterns, markers or headers in documents.

Bypass HTTPS weaknesses: You'd be forgiven for having lost patience after more than a decade of problems with HTTPS: First slow adoption by websites, then bad issuing practices by Certificate Authorities (CAs), then security breaches at CAs, then lax implementation by browsers, and finally we have trained all end-users to click blindly on "Yes" on each warning message. HTTPS interception before the traffic reaches the browser or the end-user could be used to bypass the issues with the browsers and the end-users.

Speed: HTTPS interception can be implemented relatively quickly, especially if you already have a proxy in place for outgoing Internet connections. Most products offer it as a simple add-on. Of course, the work of implementation does not end here, but making changes to endpoints, such as installing software or hardening, could take much longer and requires a lot of work.

Reasons against HTTPS interception

HTTPS interception is controversial in the IT security community for many reasons. Here are 10 good ones for skipping HTTPS interception altogether.

1. Are we serious? After a decade of telling everyone to implement HTTPS, educating users to check certificate warnings, preaching about how fundamentally important it is to encrypt network traffic, at the very moment that HTTPS is finally picking up speed, we scramble and start to intercept it, acting out the very man-in-the-middle attacks it was meant to prevent. It does not look very consistent or serious. But let's move on.

2. Strict transport security: HTTP Strict Transport Security (HSTS) is an Internet standard allowing websites to tell the browser to never accept non-HTTPS connections. This is important when a user forgets to type the S in the URL and it prevents stripping attacks. A similar thing is done with cookies: if the website sets the cookie with a secure flag, the browser will not send it back without HTTPS. These protections are important to prevent man-in-the-middle attacks. It also means you cannot just remove the SSL/TLS connection without breaking things. So intercepted HTTPS connections will have to be re-encrypted by the intercepting proxy. This is a kind of impersonation/spoofing.

Part of the deal with HTTPS interception is that connections are re-encrypted with a fake certificate, a wildcard certificate (*), which is valid for all websites. This is a kind of impersonation or spoofing, except that it is done with good intentions. The wildcard certificates are in fact not sold by real CAs, but an enterprise could create one with a non-official internal CA. This wildcard certificate then needs to be installed on all the PCs in the enterprise.

3. Whitewashing: Re-encryption with wildcard certificates effectively makes the browser and the user blind. The browser will no longer be able to warn the user about HTTPS connections and the end-user has no way to see if the certificate is valid and if the connection can be trusted. The original certificate is whitewashed.

This would not be a problem, of course, if the intercepting proxy is perfect and flawless, refusing all the bad connections and accepting only the good ones. But this is a tall order. A recent report actually shows that many of these interception products are very bad when it comes to accepting certificates, effectively opening the door to all sorts of attacks (decryption, downgrade or stripping attacks). This raises some liability questions.

4. Disrupts personal use: Social media, email providers and online banks ask their users to verify the HTTPS connection. In the case of HTTPS interception this is impossible. Maybe HTTPS interception requires some legal disclaimer about liability. Apart from the legal matters, many employees would no longer use their corporate PCs for things like social media, personal email or web banking.

In some settings this is not a problem, but I think that for most organizations it is important to allow some form of personal use of corporate PCs, for example to allow employees during their break to make a bank transfer or buy their groceries online.

5. Certificate transparency: It is not enough to re-encrypt connections with wildcard certificates. Certificate transparency is an extra protection measure allowing browsers to check if a certificate is normally used by that website. A browser like Google Chrome, for instance, warns users when a Google page is shown with a different certificate, even if it is formally valid. It is a reaction to the continuous security problems and breaches at CAs. It is an important feature and it helped discover the large-scale MITM attack on Iranian Internet users, mounted in the wake of the DigiNotar breach. So HTTPS interception requires you to also tweak the browser to accept the masquerading of the original certificate without complaints. Certificate transparency needs to be turned off.

HTTPS interception involves not only a quick intervention at the network monitoring box, but also involves changing how endpoints, browsers, and ultimately the end-users handle HTTPS connections. Choosing HTTPS interception has wider implications for the IT department at the organization.

6. Breaks with consumerization: IT in the workplace is driven by consumer products and services. Also endpoint and browsers and their security is driven by the consumer market. But because HTTPS interception has no place in the consumer market, this means that important protections like certificate transparency and certificate pinning do not work anymore in the enterprise. It would have to be tweaked or turned off on the browsers, and then implemented again on the intercepting proxy. Also, the awareness raising material, tutorials, and warning messages about HTTPS cannot be re-used anymore.

7. Disrupts BYOD: Employees are increasingly using their own personal devices in the office and for work. Sometimes these devices are used side-by-side with corporate devices. To implement HTTPS interception, personal BYOD devices need to be tweaked and configured to install and trust the wildcard certificate, and to turn off browser warnings. This not only leads to a lot of work for the IT department, which runs counter to the very idea of BYOD, but it is also likely to raise some eyebrows with users. HTTPS interception does not play well with BYOD.

8. Discourages good practices by the users: Even if we ignore BYOD, there is the problem that employees have personal devices for personal use. Social media sites and banks ask end-users to inspect certificates and to heed browser warnings. With HTTPS interception in the enterprise and no HTTPS interception at home, the employee is dealing with two worlds: At work all HTTPS connections look strange, but they are to be trusted. At home, when HTTPS connections look strange, it's an attack. This is confusing for the end-user and creates risks.

9. Limited benefits: The benefits of HTTPS interception are small and will diminish:

- Malware detection is failing. Attackers evade signature-based detection by using polymorphic malware. Sandbox-based detection is being evaded also. It is easy to see that traditional signature-based AV programs are losing the battle. It is much more important to keep systems updated then to install extra software to detect malware. Network monitoring tools cannot do better than AV (only worse, actually).
- Attackers also know how C&C traffic is detected, so they hide it. For example, there are attacks in which the C&C communications are hidden in JPG images posted on Twitter timelines. In these attacks HTTPS was not even used by the attacker! The problem is not the HTTPS encryption but the fact that there is a sea of communications to hide in.
- The same applies to exfiltration. For an attacker obfuscation is more important than encryption. If needed, attackers (insiders and outsiders) can always use an extra layer of cryptography.
- Perhaps the most compelling reason for HTTPS interception is to bypass the flaws in the browsers and the weakness of the end-user ignoring warnings. But this benefit is also diminishing, as browsers are implementing HTTPS better, and it is harder for users to ignore HTTPS warnings.

10. Hard-shell-soft-inside: HTTPS is an extension of network monitoring and detection. Investing in network monitoring and detection now is like betting on the horse that is lagging behind and is visibly tired. Not only is it easy for attackers to evade detection, it is a continuation of the traditional approach based on securing the corporate perimeter (hard-shell-soft-inside).

It is known that this approach is flawed and it also clashes with the increased mobility of staff and the uptake of cloud services. Implementing HTTPS just goes further down the wrong path.

Conclusion

HTTPS interception can provide some short-term benefits for organizations, but these benefits are limited and diminishing. More and more the emphasis is on hardening the endpoints. HTTPS also has important drawbacks. Instead of going against the wave of HTTPS uptake by tweaking how HTTPS works inside the enterprise, it is smarter to ride it and work with HTTPS as an assumption.

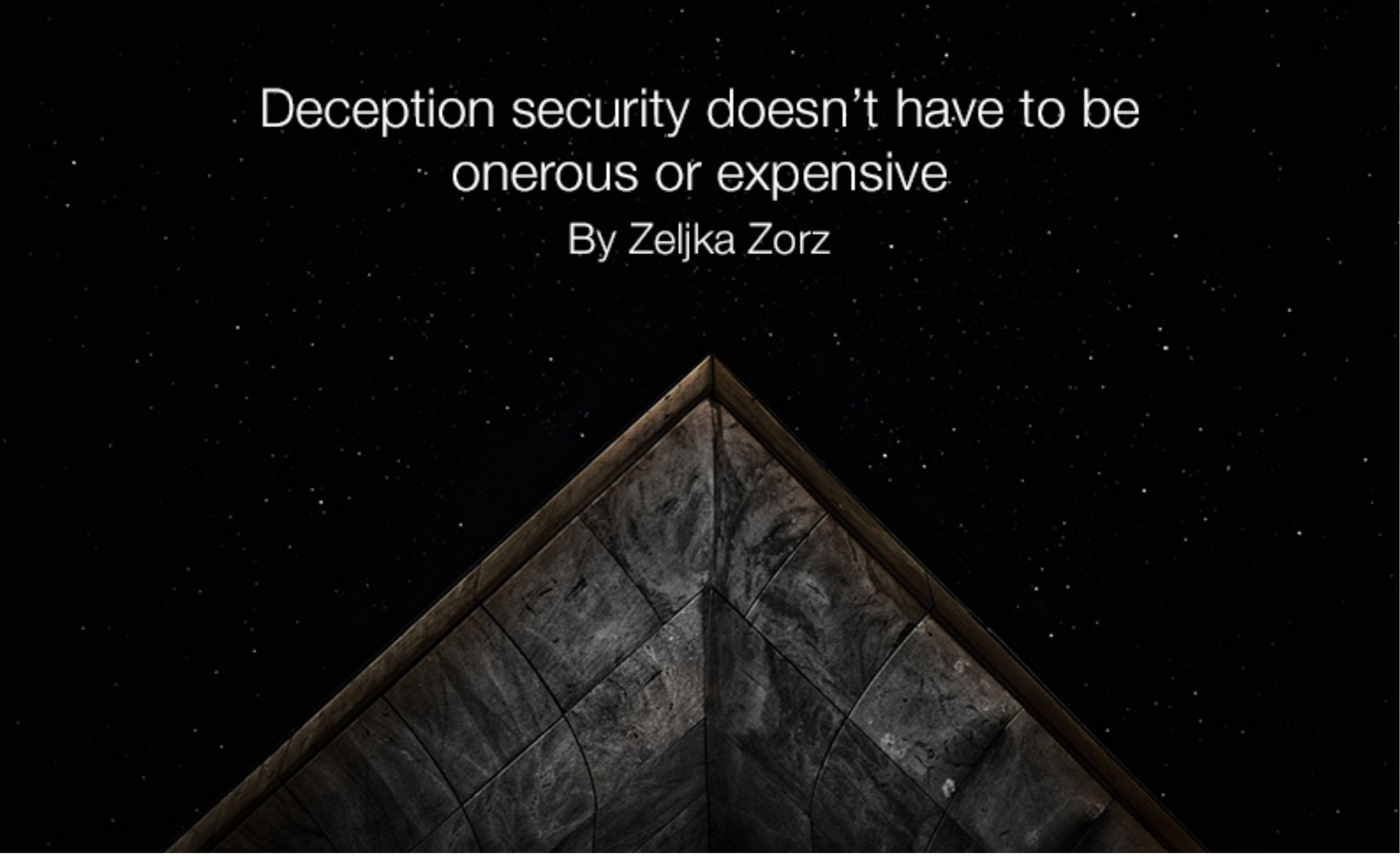
It is wiser to invest in protection measures that have a real chance against attackers, such as patching and updating, removing local admin rights from PCs, removing risky plugins, preventing users from installing software, starting detection and monitoring on the endpoint, and so on.

If you opt for HTTPS interception, make sure it is a temporary work-around and start implementing the measures needed to phase it out.

Marnix Dekker is responsible for IT Security Strategy and Policy at the European Commission.

Deception security doesn’t have to be onerous or expensive

By Zeljka Zorz



When talking about deception security, most infosec pros’ mind turns to honeypots and decoy systems – solutions that companies have to buy, deploy, and manage. But there are other ways to use deception to thwart attackers, and they do not require additional tools, pricy subscriptions, or the hiring of additional employees.

Dr. Pedram Hayati, a partner in IT security services firm Elttam who has been conducting research in the field of deceptive defense systems for years, has presented some at this year’s edition of BSides Ljubljana.

“Although deception technologies and techniques can be deployed along the entire attack chain, the attacker is most vulnerable to them in the reconnaissance stage,” he told the audience.

During his talk, Dr. Hayati demonstrated on a deceptive defence platform on Azure how a few simple configuration changes can significantly increase the cost of an attack.

He demonstrated two principles of deception security, imported from the real-world and generic enough that can be applied to any environment: the red herring (aka planting of false clues), and flooding the environment with fakes.

An attacker trying on a system will go through a lot of trial and errors, and he will be sending different payloads to the system, and the system will send back a lot of responses. Based on those responses, the attacker will change the direction for the ongoing attack, and the aim is misdirect him by offering false clues or no clues at all, Dr. Hayati noted in regards to the red herring principle.

He illustrated this by changing the configuration of a nginx web server to return random HTTP responses (200 successful, 401 unauthorized access, or 403 forbidden) when probed for particular URLs or subdomains.

The second principle involves generating a large number of fakes (open ports, services, etc.) and distributing them in different parts of the environment. The asset that the attacker is after is often rare, so making him sort through a lot of chaff to get at it can delay the progression of the attack considerably.

Deception is the most effective way to defend your assets in specific attack scenarios

To show this principle in action, Dr. Hayati first ran a port scan against a test Azure host and showed that an attacker can complete it and discover network services on it in a matter of seconds. He then opened up the first 1024 ports on the test host, configured the firewall to redirect all 1024 ports to a single port on the host, and made it to respond with a null content. Then, he ran a second port scan and service discovery against the same host.

The host responded with a seemingly never-ending list of open ports and valid services to each probe. With this simple change, he increased the duration of the attacker's service fingerprinting efforts by thirty times. "It would take an average 7 hours for an attacker to finish a basic port scan and service discovery on a single host with this setup," he noted.

For the actual commands he used and configuration changes he made, you can check out his presentation slides available at: http://bit.ly/deceptionslides

Why you should definitely think about it

"Trivial changes from the defender's side can lead to a massive increase of needed effort and time on the attacker's side, without affecting usability in any sense," he concluded, but also made sure to point out that such tweaks present just an additional security layer.

Nevertheless, deception is the most effective way to defend your assets in specific attack scenarios (e.g. the attacker has remote access to an internal host), he noted.

An attempt should be made to force attackers to spend more time and effort to figure out what is real and what is not, and make them repeatedly question whether they should proceed with the attack or not.

Zeljka Zorz is the Managing Editor of (IN)SECURE Magazine and Help Net Security (helpnetsecurity.com).

Events around the world

HITB Security Conference 2017 Amsterdam

conference.hitb.org/hitbsecconf2017ams - Amsterdam, The Netherlands / 10 - 14 April 2017

This event includes a 2-day multi-track format conference (triple track with hands-on labs), a CommSec exhibition village, Capture the Flag competition, Lock Picking Village, Soldering Village (with Mitch Altman), Hardware Hacking Village (now with car hacking by the guys who run Defcon's car hacking village and an IoT village featuring USB Armory, 44Con's HIDIOT and Michael Ossman's HackRF).

Black Hat USA 2017

www.blackhat.com/us-17/ - Las Vegas, USA / 22 - 27 July 2017

Now in its 20th year, Black Hat is the world's leading information security event, providing attendees with the very latest in research, development and trends. Black Hat USA 2017 kicks off with four days of technical Trainings (July 22-25) followed by the two-day main conference (July 26-27) featuring Briefings, Arsenal, Business Hall, and more.

Report: BSides Ljubljana 0x7E1

By Zeljka Zorz

This year marked the first time that the Help Net Security team attended BSides Ljubljana. As it happens, it was also my first time at a BSides event, and I was not disappointed.

The venue (Poligon creative centre) was great - big enough for the talks and the CTF tournament, and small and compact enough to allow a constant intermingling of all 170 or so participants, the speakers and organizers.

The organizers did a top-notch job attracting good speakers and formulating a well-balanced, two-track talks schedule, as well as keeping everything smoothly rolling according to it.

The only problem for us was to choose which talk to attend, when both tracks were full with interesting topics. Should we learn more about IoT hacking? Or security analysis of binary applications? How about cookie stealing and session hijacking? How to run a Security Operations Center in Python on top of ElasticSearch? Deception defense? Micropatching? Password hashing? APT attacks against government agencies? Some tough choices had to be made.

Personally, the talk - and ensuing discussion - that still pops up daily in my head was the one about responsible disclosure and ethical hacking by Gorazd Božič, the Head of the Slovenian national Computer Emergency Response Team. Boy, did he give all of us a lot of material to chew on!

But the best thing about an event like this is seeing old friends and making new contacts. Bonding over a discussion on Twitter is one thing, but meeting those people and exchanging knowledge and opinions face-to-face is even better.

5 spring cleaning tips for your Identity and Access Management program

By Kayne McGladrey

Spring cleaning is a tradition for millions of families, but most companies lack the same tradition when it comes to the long-term management of their Identity and Access Management (IAM) programs. This is not benign neglect, but rather an underlying fear that the IAM program resembles a shaky tower of cardboard boxes full with random stuff, sitting in the garage.

It is tolerable to look at infrequently, but you will need to sign a change control order before adding another box containing a singing fish, an indoor grill, and a spinning mop to the pile.

Often, companies only uncover these tangled messes and curious provisioning logic rules when considering upgrading or replacing their IAM vendor's product. In this article, we will examine how to clean up the five most common messes you will find as you air out the storage unit of your IAM program.

A useful annual technique for assessing individual components of IAM programs is ADR:

Approach: Interview the personnel most knowledgeable about each portion of the IAM program, and have them explain how and why it works.
Deployment: Ask the staff you interviewed to provide you with the written documentation describing their portion of the IAM program.
Results: Ask them to show you the results of a live transaction. This might be adding a user to an application, or privilege revocation. Check that the results match what was verbally described and the written documentation. Often, people will apologize at this point, explaining that either the documentation is out of date, that they forgot to mention a step, or that there's now a manual step required to achieve the desired end state.

The gulf between the results and the approach or deployment are a leading indicator of trouble spots such as these:

The provisioning rules refer to outdated business processes

A shipping company that I worked with in New Mexico had an elegant provisioning system with several levels of approvals when a user requested access to production systems. Emails were routed to appropriate personnel, approvals were granted or denied, and the request would then be processed automatically. At least, that is how it was supposed to work.

The company had several reductions in workforce over the years, and this resulted in approval emails being routed to managers and VPs who were no longer with the company. The emails would bounce, and this had initially caused chaos in the approvals process. However, the IAM provisioning lead had developed a workaround: instead of finding new approvers or disabling the invalid approvers, they had modified the provisioning logic to accept an email "bounce" notification as an approval.

This curious implementation was not done maliciously – it was put in place when no one on the IAM program knew who would be responsible for approving access to production after the workforce reduction. But the change had potential regulatory compliance consequences, as the requestor would be granted production access with no human oversight if all three emails in an approval bounced.

The solution was to first identify the right managers and VPs for production-level access approval. This was done through a series of meetings before anyone sat down to modify the provisioning logic. Additionally, the logic to accept a "bounce" email as an acceptance was changed, and they began to be treated as a rejection.

There are multiple duplicate and identical rules

A colleague of mine was on a project with a media company in Southern California. As part of a planned consolidation of their IAM program, they envisioned importing authorization rules from their privilege systems into a centralized repository.

This was such a common project at the time that my colleague had built some automated scripts to expedite the import process. What she had not considered was the implications of a client that did not understand using groups for managing privileges.

The client had a set of 63,000 authorization rules, many of which were rather granular, such as the ability to reboot a production server. But instead of creating a group of servers and a group of users, the client had instead created a rule for each user on each server. That meant that every time they hired a new system administrator, they had to add about 500 new authorization rules to the set.

The solution to these excessive rules was to review the privileges that were granted to each user to find common elements. For example, it was a standard privilege that a systems administrator could install updates on the systems. Identifying the standard role of "systems administrator" was essential, because then additional privileges could be assigned to that role, such as restoring files from backup or rebooting a system.

After much analysis, four primary roles were defined. The result of consolidating the roles and privileges was that it was considerably easier to produce audit reports showing who had privileges across the computing estate.

The rules refer to outdated systems

A pharmaceutical company that I worked with in the Northeastern United States had a project to centralize user authentication and authorization under a single source. One of the findings during the analysis phase was that this company was using Network Information Service (NIS) netgroups to control access to file shares on dedicated file-sharing appliances. By itself, this was not surprising, as NIS and file sharing were conventional technologies years ago. The surprising thing was that the client's team had decommissioned the use of NIS, and the dedicated file-sharing appliances were long gone. The provisioning system would throw an error when attempting to edit the NIS netgroups file, and the development team had written error handling code that discarded the error.

Migrating from one IAM product to another is a good time to identify and remove references to systems that no longer exist. However, the best practice when shutting down a component of an IAM program is to ensure that all connectors are disconnected, and all programmatic logic is disabled. Keeping those around just adds operational complexity for the next team responsible for the IAM program.

No one understands the security implications of the rule

When I was offered the opportunity to work on a project for three weeks in Amsterdam, I accepted, thinking the hardest challenge would be finding a good cup of black coffee with no specials.

The client was a financial services firm that had a permissive stance on security that had led to several breaches and unauthorized trades in rapid succession. They were considering starting an IAM program so that they could limit user access. Before arriving, I'd asked them to produce a list of users for each system, so that we could review who had access to critical business systems.

When I arrived, they explained that the traders on the floor logged in with the default user account on Linux and then opened their trading program. Unfortunately, the default user account was named "root." For convenience, it was the same password on all systems, and it was posted on signs near the terminals in case new hires forgot. (If you are unfamiliar with Linux, the root account is the equivalent of the Administrator account on Windows).

The trading firm had initially started on Windows terminals, before hiring a talented team of Linux developers who helped build and maintain an efficient trading platform. The development testing was conducted as the root account, and consequently, all documentation for the system indicated using the root account to log in to the production systems. As the IT staff at the financial services firm had a background in Windows, they saw no initial concerns with this operational model, as they did not understand the significance of the root account.

Although there was no quick solution, the project was based on having users log in with their own credentials. They already used Windows computers with distinct logins for access to their office software and their email. After several meetings, the CIO understood that the same model could be successfully applied to the Linux trading environment with no loss of functionality and a considerable improvement to security.

The longer-term solution was to create a dedicated security analyst role within the firm so that all new systems could be examined for potential security implications.

NO ONE UNDERSTANDS THE SECURITY IMPLICATIONS OF THE RULE

Additional manual steps are required

Software vendors will periodically mark their software as "end of life." While the original vendor may offer an upgrade path and competitors may offer competitive migration tools, not all customers migrate. At Integral Partners, we have encountered numerous clients over the years whose IAM programs include software that's no longer covered by extended maintenance and has been abandoned.

One client we worked with was a manufacturing company in the Midwestern United States who used an LDAP Directory and an Identity Management platform from a vendor that had ceased operations. Although the original hardware and software was still running, they had developed a duct-tape patchwork of scripts and programs that ran after each provisioning operation.

Most of these were to provision systems that did not exist when the original vendor had stopped supporting their platform, though some were to keep the older software running. Many of the scripts were undocumented and had been written and modified by multiple authors, further increasing the complexity.

The solution in cases like this is to take an inventory of which systems need to be provisioned, which roles and privileges need to be assigned, and who's responsible for approvals. Next, build the same logic, transactions, and approvals on a new system that can connect to lab instances of the endpoints being provisioned.

As each transaction runs in production, have it also run in the lab to confirm that there's parity in the end state on each transaction, whether it is for granting a privilege or for removing a user account.

The next step is the most difficult – stop running operations through the old system and start running all IAM transactions through the new system, and wait to see what was missed. It often takes days to find an unusual edge case scenario that the team had not initially identified and migrated to the new IAM system.

Only after the new system has been operational for about six months with no "missing" transactions should anyone consider decommissioning the legacy system, as the advantage of leaving it running ensures that it will be accessible as a reference point for unusual IAM transactions.

The alternative would be to review and attempt to port the spaghetti code from the older platform. However, this carries the risk of porting logic errors and hacks that were put in place solely to sustain the older software. Competitive migration toolkits from vendors often perpetuate poor design decisions and should be approached warily.

SPRING IS A GOOD TIME FOR ORGANIZATIONS TO START AN ANNUAL TRADITION OF REVIEWING THEIR IAM PROGRAM FOR JUNK THAT CAN BE REMOVED

Although this is not an exhaustive list, the five IAM disarrays described in this article are the most frequently encountered today. Spring is a good time for organizations to start an annual tradition of reviewing their IAM program for junk that can be removed.

The Approach – Deployment – Results method will help open the storage bins and find potential areas for improvement, as well as unmanaged risks to the IAM program. Applied regularly, this approach can reduce the possibility of clutter and keep your IAM program running smoothly for years to come.

Kayne McGladrey is an IEEE Member and Director of Information Security Services at Integral Partners.